SCS STUDY REPORT - VOLUME 1
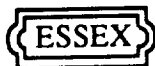
SPACE STATION
SIMULATION COMPUTER SYSTEM (SCS) STUDY
for
NASA/MSFC

OVERVIEW and SUMMARY

TRW-SCS-89-T7

31 October, 1989

TRW

ESSEX

GRUMMAN

TRW

TRW System Development 213 Wynn Drive
Division Huntsville, AL 35805
Defense Systems Group 205 837 2400

SN51946.000
G002.VOB.89-3312
31 October 1989

National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Marshall Space Flight Center, AL 35812

Attn: Mr. M. Naumcheff/EL12

Subject: Contract No. NAS8-37745/P00002
Final Technical Report
Simulation Computer System for Space
Station Program

In accordance with the requirements of the subject
contract, the final technical report titled SCS
Study Report, consisting of six volumes is herewith
submitted and distributed as shown.

TRW Inc.
Defense Systems Group

V. O'L. Bain
Contracts Administrator
Huntsville Operations
System Development Division

Distribution:

CN22D (5)
AT01 (1)
CC01 (1)
EL 12 (20)
KA02 (1)
NASA Scientific and Technical Information Fac. (2)

SPACE STATION SIMULATION COMPUTER SYSTEM (SCS) STUDY

OVERVIEW and SUMMARY

SCS _____ SCS

TRW

(ESSEX)

GRUMMAN

SCS _____ SCS

CDRL: TRW-SCS-89-T7

31 October, 1989

Approved By:

J H Jensen

J. H. Jensen, Manager
SCS

Released By:

[signature]

P. G. Dombrowski, Manager
Huntsville Systems Laboratory

SCS _____ SCS

Under:

Contract No. NAS8-37745/P00002

Prepared for:

**NASA
MSFC EL12
Huntsville, Alabama 35812**

Prepared by:

**TRW System Development Division
213 Wynn Drive
Huntsville, Alabama 35805**

LIST OF FIGURES

INTRODUCTION

NASA's Space Station Freedom program (SSFP) planning efforts have identified a need for a payload training simulator system to serve as both a training facility and as a demonstrator to validate operational concepts. The envisioned MSFC Payload Training Complex (PTC) required to meet this need will train the Space Station payload scientists, station scientists, and ground controllers to operate the wide variety of experiments that will be onboard the Space Station Freedom. The Simulation Computer System (SCS) is the computer hardware, software, and workstations that will support the Payload Training Complex at MSFC.

The purpose of this SCS Study is to investigate issues related to the SCS, alternative requirements, simulator approaches, and state-of-the-art technologies to develop candidate concepts and designs. This study was performed August 1988 to October 1989, thus the results are based on the SSFP August 1989 baseline, i.e. pre-Langley configuration/budget review (C/BR) baseline. Some terms, e.g. combined trainer, are being redefined.

This SCS "Overview and Summary" presents an overview of the study activities, and a summary of study results. The "Overview and Summary" is Volume 1 of the SCS study report, and presents a road map to the various volumes that make up the complete SCS Study Report.

All of the reports issued as part of the SCS Study are products of specific SCS tasks. These reports have been updated and are contained in the study report as separate volumes. To make the study report easy to use as a reference, the volumes are arranged with the most current data first, i.e. the reports issued at the first part of the study are the last volumes. The volumes are:

Vol.#	Title	Output from SCS Task
1.	Overview and Summary	7 - Study Report
2.	Concept Document (candidate requirements)	3 - Develop Requirements
3.	Refined Conceptual Design Report	5 - Refine Designs
4.	Conceptual Design Report	4 - Conceptual Designs
5.	Study Analysis Report	2 - Parametric Analysis
6.	Study Issues Report	1 - Study Issues

Volumes 1 to 3 are expected to be of current interest, while volumes 4 to 6 are included for historical purposes so that work once done can be referenced and not redone. The SCS Study effort consisted of 7 tasks. These are each discussed in the following pages.

TASK 1 - Identify and Analyze SCS Study Issues

This was the first task to be performed. First, issues were identified from all sources. These included the NASA Statement of Work (SOW), the TRW proposal, and working groups which focused the experience of NASA and the contractor team performing the study - TRW, Essex, and Grumman. A total of 20 training issues and 14 associated development issues were identified. Overall analysis of all the issues was begun by creating a list of all the operational functions for which the SCS could be used. These operational functions evolved into three major areas:

1) Training Functions - Doing the required training

2) Development Functions - Developing training simulators and aids

3) Operations Evaluations - Evaluating operations and technology to support training

To ensure all the operational functions were valid, a matrix of SCS users vs. SCS operational functions was created. SCS users are:

1) Flight Crew

2) Ground Support Crew

3) Science Users

4) Operations Planning Personnel

5) PTC Personnel

To help ensure that all relevant issues were identified, a matrix of issues vs. operational functions was created. A different view of the issues resulted form creating a cost factor vs. SCS issue matrix. All these matrices aided in beginning the detailed analysis of each issue, and are contained in Volume 6 "Study Issues Report".

Next, a standard form was created to capture the results of the analysis of each individual issue. Analysis and discussion of each of the issues continued resulting in resolution of 13 of the training issues, and 6 of the associated development issues. All of the issues identified are listed in Figure 1. The unresolved issues (shown by a yes under the Further Study heading in Figure 1) were the subject of the SCS Task 2 effort.

The details of the analysis and results of SCS Task 1 are captured in Volume 6 "Study Issues Report". Assumptions are clearly listed as a part of each issue study in this report and are collected and included at the end of this "Overview and Summary".

Training Related Issues

Further Study?	Issue Number & Title
Yes	T-1. Scope of Payload Crew Training in PTC
No	T-2. Scope of Ground Operations Personnel Training in PTC
Yes	T-3. Scope of OMS Training in PTC
Yes	T-4. Scope of Integrated Core Subsystem Training in PTC
No	T-5. Fidelity of SS Payload Subsystems Simulations
No	T-6. Fidelity of SS Experiment Simulations
No	T-7. Fidelity of SS Experiment to System Interfaces
No	T-8. Fidelity of SS Internal Data Flows Simulations
No	T-9. Fidelity of SS Downlink and Uplink
Yes	T-10. Fidelity of Element Control Workstation (ECWS)
No	T-11. Support for Training Multiple Missions Simultaneously
No	T-12. Support for Integrated Simulations with Other NASA Centers
Yes	T-13. Support for Interoperable (Remote Executions) Simulations
No	T-14. Requirements for SCS Interface with External Facilities
Yes	T-15. Requirements for PTC Payload Video Data
No	T-16. Requirements for Simulation Parameter Update Rate Requirements
No	T-17. Requirements for High Rate Data Requirements
No	T-18. Requirements for Virtual Instruments
No	T-19. Requirements for Simplified Simulator Operations Setup and Control
Yes	T-20. Support for Onboard Training

Associated Development Issues

Further Study?	Issue Number & Title
No	A-1. Utilization of SSE Capabilities
No	A-2. Techniques for Integrating and Maintaining PI-Provided Simulators
Yes	A-3. Techniques for Supporting late changes to simulators
No	A-4. Allowing Software Transportability between SCS and other centers
Yes	A-5. Techniques for Integrating Flight Hardware/Software with SCS Simulators
Yes	A-6. Flexibility for Allowing Advanced Technology Insertion
Yes	A-7. Implications of Simulation Development Cycle
Yes	A-8. Sizing Growth Potential in Capability/Capacity
No	A-9. Defining Telemetry Data Format and Calibration
Yes	A-10. Fidelity of DMS Interface
Yes	A-11. Definition of "No single point of failure"
No	A-12. Requirements for Interfaces with SOAN and SSIS
No	A-13. Requirements for Configuration Management of Simulation Software
Yes	A-14. Definition of GSE-Provided Services

Figure 1.   List of SCS Study Issues

TASK 1 RESULTS SUMMARY

Although the results of each study are important and had an effect on the candidate requirements, the key Task 1 results can be summarized as follows:

a) Almost all the simulator fidelity issues were resolved, and they were resolved in favor of realistic training and fairly high fidelity in the simulators (Issues T-5 through T-9 and T-16 through T-18). This is consistent with the level of Spacelab training currently provided in the MSFC Spacelab Payload Crew Training Complex (PCTC).

b) Many of the interface issues were resolved, and these were resolved to require the maximum use of the Space Station Software Support Environment (SSE), and thus maximize and promote portability of the SCS simulations for use at other centers, and importation and utilization of models needed by the SCS from other centers (Issues T-12, T-14, A-1, A-2, A-4, A-9, and A-12).

c) Assumptions were made to allow the SCS NASA/contractor team to complete the study. As these are changed, the analysis results and subsequent design must be changed. The analysis assumptions are clearly stated as a part of each Task 2 study, and are summarized at the end of this "Overview and Summary".

TASK 2 - Perform Studies and Parametric Analysis

This second SCS task utilized parametric analysis, graphical analysis, and study of available information, including historical data, reference missions, vendor information, and Space Station Freedom Project reviews and documents, to resolve the issues unresolved in Task 1. All the issues shown in Figure 1 with a yes under the Further Study heading were resolved by this effort. However, the following caveats apply:

1) The resolution of issues is only as good as the external data input to each analysis. What was used as input to each study is clearly stated under Inputs, part of the form developed for these analyses. Details of parts of the Space Station Freedom Program (SSFP), especially the Data Management System (DMS) and Software Support Environment (SSE), were unavailable or changing rapidly during the study. The implication is that as the input data is updated and changes, different results will be obtained using the same analysis.

2) Assumptions were made to allow the SCS NASA/contractor team to complete the study. As these are changed, the analysis results and subsequent design must be changed. The analysis assumptions are clearly stated as a part of each Task 2 study, and are summarized at the end of this "Overview and Summary".

TASK 2 RESULTS SUMMARY

Much of the analysis done to resolve the remaining open issues under Task 2 was directed at defining the scope and load that the SCS must support. Reference missions and historical SpaceLab data were used to determine the details, but the key, top level analysis was done graphically based on SpaceLab and other simulator and training experience. The results of this are shown on Figure 2 PTC Training Increment Flow Requirements.

Figure 2 shows the types of payload training and the load this training will impose on the SCS. The vertical lines represent SSFP launches, and are 90 days apart. Launch number 1 is the SSFP First Element Launch (FEL), and will be an assembly launch with no payloads included. The first launch to include payloads is launch number 4.

To support this schedule, the classroom work and computer based training (CBT) for launch 4 (Increment 4) must begin 18 months prior to the launch. During the first 6 months of PTC training, it is envisioned that both flight crew and ground controllers will spend some of the time training at Principal Investigator (PI) sites.

Legend

- - - - - - PTC Class/CBT Training
PTC Individual P/L Training
PTC Combined P/L Training
PTC Consolidated Increment Training
PTC POIC Training
SSTF Training
Consolidated Payload Simulation

Launch Number

Flight 4

Flight 5

Flight 6

Flight 7

Flight 8

Flight 9

Flight 10

Increments are assumed to be launched on 3 month centers. First payload launch is 4th.

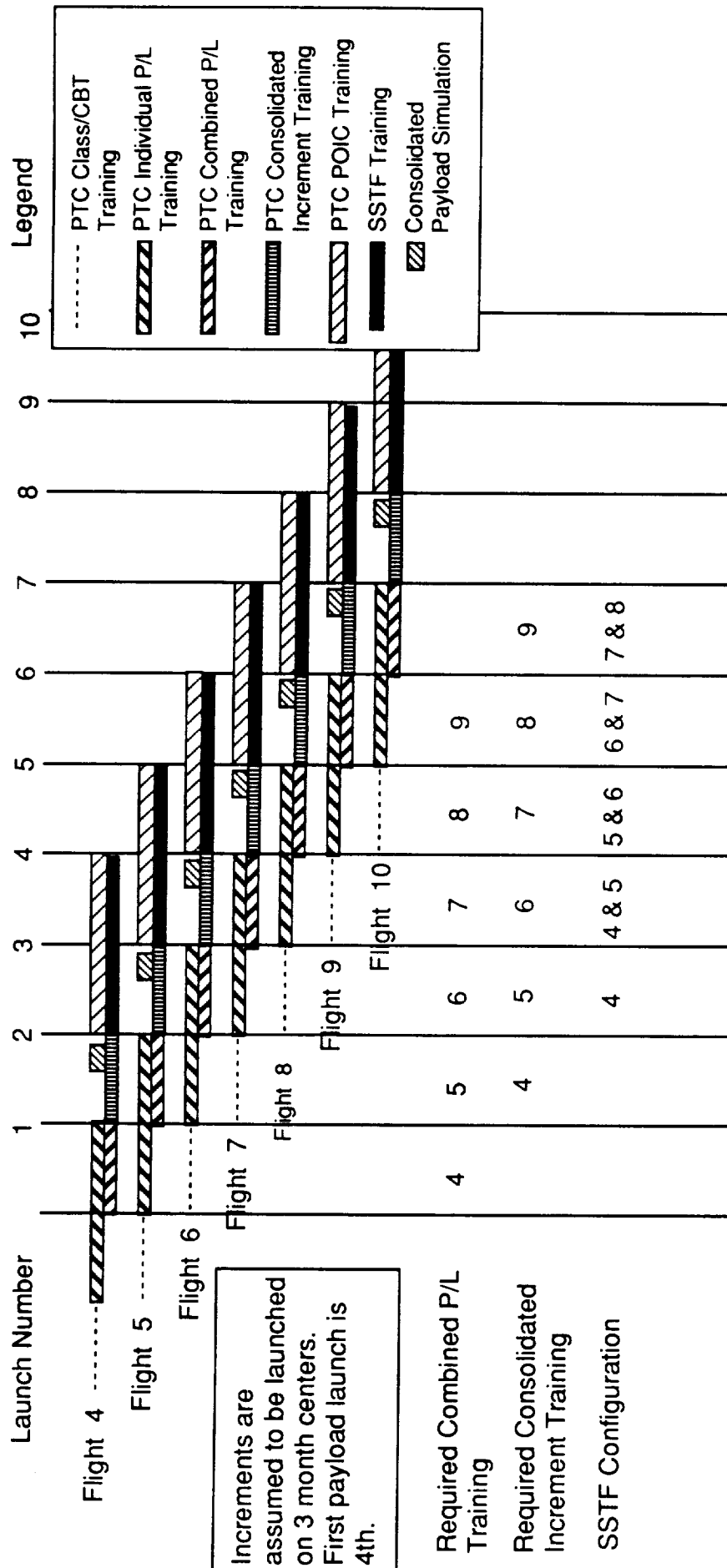|									
Required Combined P/L Training	4	5	6	7	8	9			
Required Consolidated Increment Training		4	5	6	7	8	9		
SSTF Configuration		4	4 & 5	5 & 6	6 & 7	7 & 8			

Figure 2 . PTC Training Increment Flow Requirements

The classroom/CBT training will be followed by 6 months of training on individual payloads. Individual payload training will be accomplished on part task trainers (PTTs). These are expected to consist of a crew interface which controls the payload, a rack mounted payload representation, and an instructor station. During the last 3 months of this period, the crew will train on both individual payloads and the payloads combined (physically and electronically) into individual laboratory complements (US Lab, Columbus, and JEM). This is the Combined payload training shown on Figure 2.

Consolidated increment training will be the focus for the next 3 months. The crew will train on all the payloads in a consolidated increment, i.e. all the payloads they must operate when they serve their first 90 days of their 180 day tour onboard SSF. All three labs must be physically and electronically integrated to provide this Consolidated Increment training. Consolidated payload simulations will also take place during this period. These consolidated payload simulations will train flight crews at the PTC with trainees at the Payload Operations and Integration Center (POIC) and other operations centers, i.e. Regional Operations Centers (ROCs), Discipline Operations Centers (DOCs), and User Operations Facilities (UOFs).

The next 6 months (the last 6 months before launch), the crew will train primarily at the Space Station Training Facility (SSTF) at JSC on all the systems onboard, i.e. both core systems (those key to operating and controlling the station) and payload refresher training.

From Figure 2, it is clear the SCS must support crew training on a minimum of 4 different increments simultaneously. This includes as a minimum one full consolidated payload increment, one combined payload configuration, and individual payload training on payloads from 2 other increments.

From the baseline load shown in Figure 2, further analysis (based on SpaceLab data and the contractor team's simulator development experience) was done to yield loads for the development effort. This showed that the SCS development system must support 105 simulator developers (plus or minus 20 percent). The load required to support the operations evaluation function was determined to be negligible, as much of this function would be performed as part of the training effort.

TASK 3 - Develop and Present SCS Requirements

Work on this task began as the Task 2 effort was drawing to a close. First, a methodology for developing and format for presenting the candidate requirements was selected.

The methodology selected was structured analysis, and the tool selected to accomplish this was the SSE selected/approved PowerTools. The PowerTools suite of tools supports the design process from front-end requirements analysis to software development. The PowerTools initial individual tool used in requirements analysis and development is FreeFlow. FreeFlow supports the Yourdon/DeMarco method of structured analysis, producing data flow diagrams in the standard format, and linking

and checking input and output data flows. All of the FreeFlow data flow diagrams and input/output lists are included as part of the candidate requirements.

The SSFP System Specification Data Item Description (DID) was selected as the format for presenting the candidate requirements. This provided a good basic structure for all the candidate SCS requirements. Some small additions were made to the DID outline to suit the SCS needs.

Then began the structured analysis and writing of requirements. This work depended upon and heavily utilized the results of the first two tasks. Additionally, many other system level specifications were used as guidelines.

The SCS Concept Document (Volume 2) contains these candidate requirements, which were developed from the study and analysis efforts of SCS Study Tasks 1 and 2, and from the Yourdon/DeMarco structured analysis method as supported by the SSE PowerTools.

TASK 3 RESULTS SUMMARY

The SCS concepts and candidate requirements developed during the SCS Study Task 3 reflect the need for a high-fidelity, real-time training system that produces operationally ready flight and ground crews. The SCS concepts and candidate requirements also reflect the PTC/SCS Training Objectives shown in Figure 3. These objectives were developed working closely with NASA training personnel during SCS Study Task 3 and are grounded on SpaceLab training experience. The SCS concepts and candidate requirements must also support the needed PTC/SCS Interfaces to other SSFP Elements shown in Figure 4 and support the flow of trainees shown previously in Figure 2.

To accomplish the required training, the SCS Study has determined that the PTC/SCS will consist of the components shown in Figure 5. These components are:

POIC Consoles - These are functionally, as well as cosmetically, equivalent to the actual consoles used in the POIC. The screen displays and interactions will be exactly like the operational consoles, but the actual hardware may be different. In the current baseline, these consoles will be part of and located in the POIC. There will be no separate set of consoles in the PTC/SCS.

Consolidated Increment Trainer - This consists of all three labs (US, Columbus , and JEM), connected together via the 2 connecting nodes. A logistics module is also included. It also includes the payload racks, power, lights, Data Management System (DMS) Kits and their international partner equivalents, and other hardware needed for realistic simulation of the entire laboratory portion of the SSF.

Training Type	Objective
Computer Based Training	Will train individual students utilizing scenarios without instructor intervention -- basic use will be to provide preliminary or introductory instruction via screen text, video, and graphics combined with questions to which the student would respond.
Part-Task Training	Primarily for developing single crewmember operating skills associated with individual payload flight operations. Will also be utilized for the development of ground support personnel operating skills associated with individual payload operations.
Combined Training	Primarily for training a team of 2 or more crewmembers to operate multiple payloads combined into specific labs. Supports the combination of crewmembers and ground support personnel for training on payload operations specific to a lab.
Consolidated Training	Primarily for training 4 or more crewmembers located in Freedom modules or the combination of crewmembers and ground support personnel for training on integrated payload operations throughout the entire manned base.
SSTF Integrated Training	Allows a student team to train on an entire mission increment at JSC with payload simulators running in a full-scale mode with the SSTF.
Consolidated Payload Simulation	Purpose is training crew at the PTC with teams of students at other operations centers, including the POIC and user operations centers (ROC's, DOC's, and UOF's) on specific flight increment objectives including reworking the short term plan, payload operations and updates, interactions with telescience operators, shift handovers, and payload malfunctions.
POIC Training	POIC Cadre members and certain representatives of remote operations centers can train on POIC systems, protocols and procedures using a representative subset of POIC components.

Figure 3. PTC/SCS Training Objectives
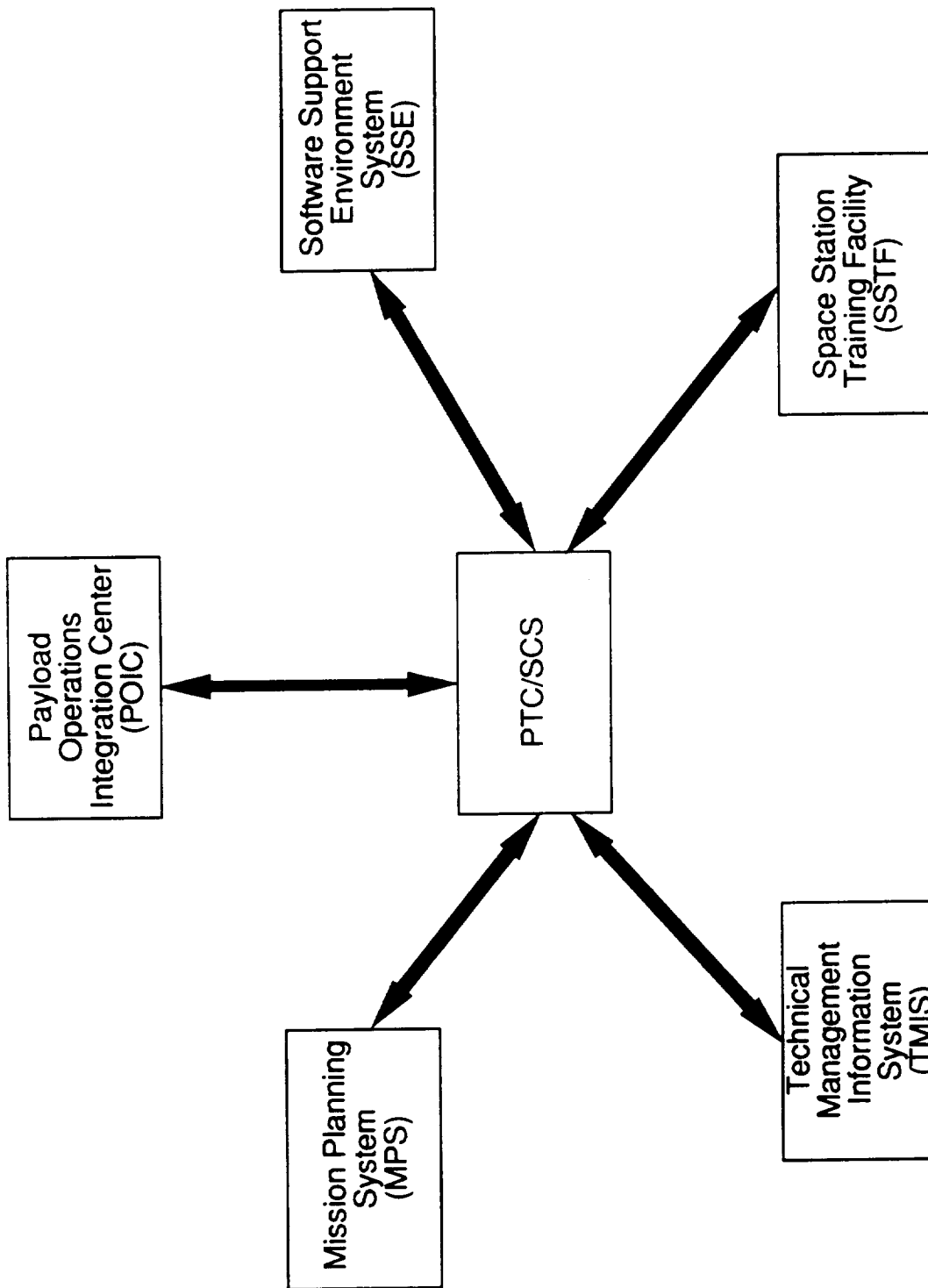
Figure 4. PTC/SCS Interface to other SSFP Elements

SCS Control
Environment

Instructor
Stations

Training
Session
Manager

Ground
Support
Equipment

CBT
Stations

Consolidated
Increment
Trainer

Combined
Trainer

Attached
Payload
Trainer

w/DMS
Kit

Part
Task
Trainer A

w/o
DMS
Kit

Part
Task
Trainer B

POIC
Trainers

Development & Test
Environment

Development
Facility

IT &V
Facility

External
PTC Interfaces To:

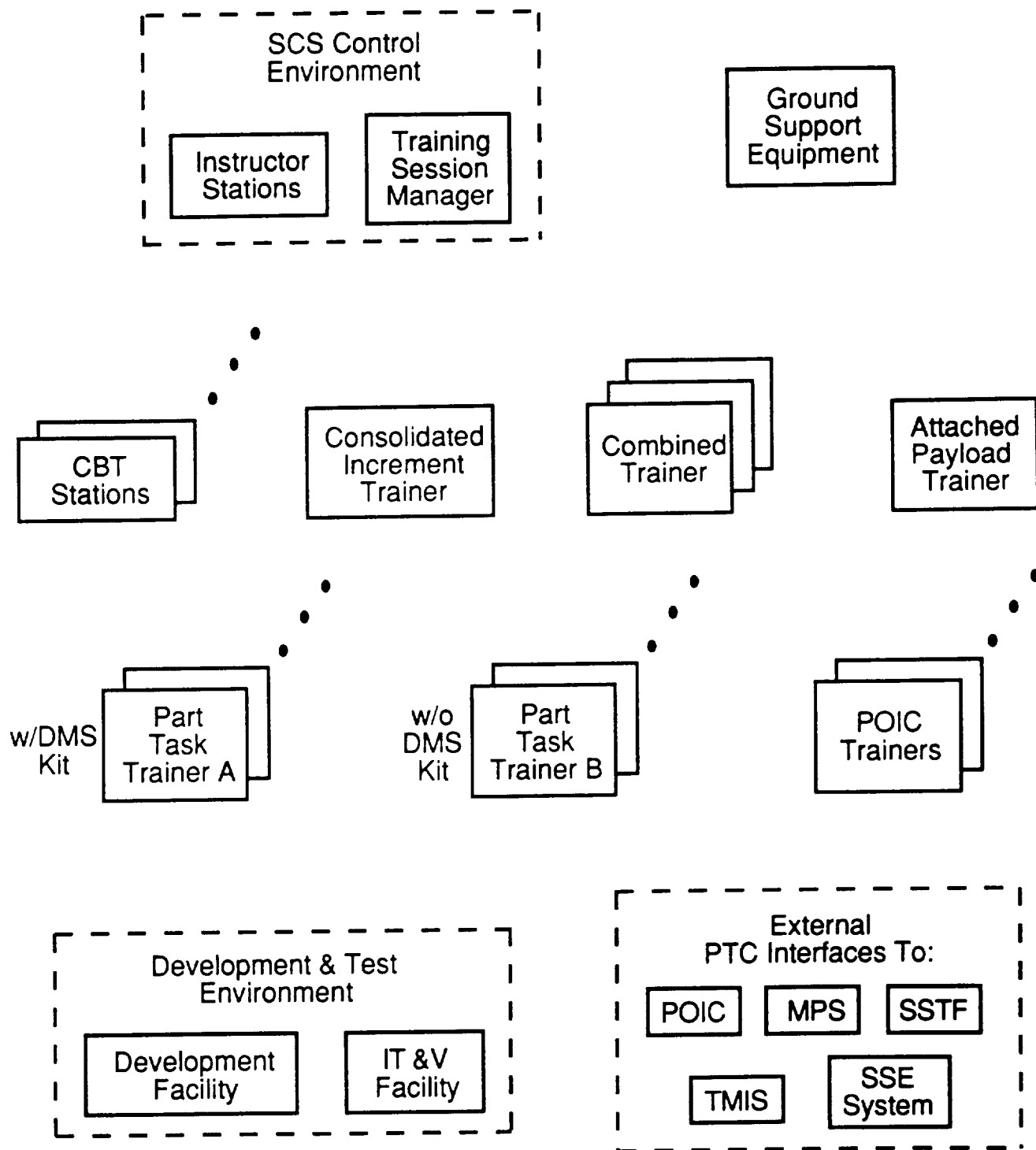POIC

MPS

SSTF

TMIS

SSE
System

Figure 5. PTC/SCS Components

Combined Trainer - This consists of individual, stand-alone labs (US, Columbus , and JEM), with a portable node and a portable logistics module. These also include the payload racks, power, lights, DMS Kits or their international partner equivalents, and other hardware needed for realistic simulation of the individual laboratories.

Attached Payload Trainer - This consists of the C&D panels, screens, and controls which make up the crew station for payloads that are attached on the truss outside the Labs. This crew station will be in one of the SSF nodes.

Part-Task Trainers - Each of these consists of one to three payload racks and a console or workstation to control the simulation of individual payloads.

Computer Based Trainers (CBT) - These consist of stand-alone consoles or workstations for individual student training.

Simulator Development Facility - This consists of computers, software tools, and environments to aid in simulator development.

Integration, Test, and Verification Facility - This consists of all the hardware and services needed to support the PTC/SCS in integrating, testing, and verifying payload simulators.

External PTC Interfaces - These are the SCS components for communicating to the external elements as shown in Figure 4.

SCS Control Environment - This includes the instructor stations which the instructors use to control training, and the training session manager which will control the SCS configuration and prepare (initialize) the SCS to be operated.

Ground Support Equipment Subsystem - This consists of all the hardware and services needed to support PTC/SCS use of payload flight equivalent hardware such as facility power, facility heating and cooling, and facility audio/video.

The SCS Summary Context Diagram (Figure 6) shows the top level SCS System Functional Flow. Creating a diagram like this is the first step of the Yourdon/DeMarco analysis methodology. This summary context diagram graphically delineates the functional boundary (the domain) of the SCS system, and the external environment. The data flows (arrows) on the summary context diagram depict the communication of data between the system and its external sources and receivers of data. The circle in the center of the diagram represents the SCS system, and the rectangles represent entities external to the SCS, including hardware that is part of the PTC. The following list explains each external entity:

- Mission Planning System (MPS) -- represents the MSFC system that will provide all the information needed to most efficiently operate the payload missions. This information includes such things as timelines, orbital ephemerides, and SSF orientation data.
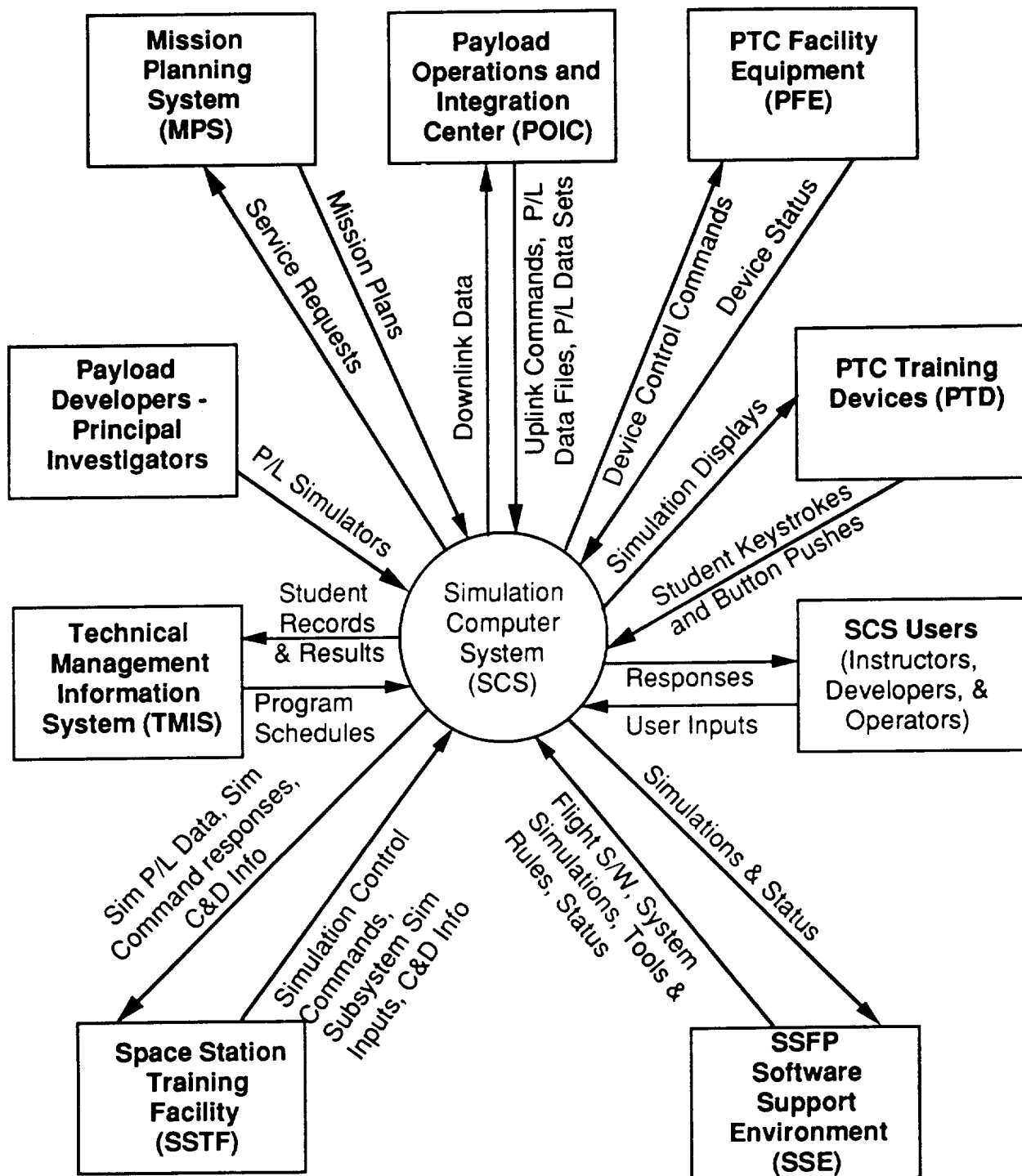
Figure 6. SCS Overview Context Diagram

- Payload Developers - Principal Investigators -- represents developers who do not use the SCS to develop payload simulators, e.g. PIs who develop simulators at their sites.

- PTC Facility Equipment (PFE) -- represents PTC equipment which are not direct training devices, specifically Ground Support Equipment (GSE) and audio/video systems such as facility VCRs and cameras.

- Payload Operations and Integration Center (POIC) -- represents the Payload Operations Integration Center that will interface with the SCS to send and receive payload downlink and uplink data for payload crew and POIC cadre training.

- PTC Training Devices (PTD)-- represents the hardware (control and display panels and crew workstations) in the consolidated, combined, and part-task trainers that simulates the real onboard payload operations hardware. This simulated hardware is where the students will interface with the SCS to receive their high fidelity payload operations training.

- Software Support Environment (SSE) -- represents the Software Support Environment System that will provide all tools, rules, and procedures for the SCS and will also provide system simulations, flight software, and configuration management status information. The SCS will send the SSE simulations and necessary configuration management information.

- Space Station Training Facility (SSTF) -- represents the Space Station Training Facility at Johnson Space Center. The SCS will be required to support the SSTF in full integrated mode on-site at JSC.

- Technical Management Information System (TMIS) -- represents the Technical Management Information System that will provide the SCS with program schedule information and will store student records and results.

- SCS Users -- represents the instructors, developers, and operators, i.e. all the people who will use the PTC/SCS to provide training for the students.

Task 4 - Develop SCS Conceptual Designs

Once a candidate set of requirements was developed, candidate top level conceptual designs to meet these requirements were developed under this SCS task. This task explored the spectrum of possible SCS top level architectural designs.

In the first step of this task, a methodology was developed to ensure that all relevant design dimensions were addressed, and that all feasible designs could be considered. The development effort yielded the following method for generating and comparing designs in Task 4:

1. Extract SCS system requirements (functions) from the concept document.

2. Develop design evaluation criteria.

3. Identify system architectural dimensions relevant to SCS designs.

4. Develop conceptual designs based on the system requirements and architectural dimensions identified in step 1 and step 3 above.

5. Evaluate the designs with respect to the design evaluation criteria developed in step 2 above.

The "SCS Conceptual Design Report" (Volume 4) contains the detailed results of the five steps of Task 4.

TASK 4 RESULTS SUMMARY

The designs developed under this task are the key result. A broad spectrum of top level system designs were developed. A spectrum of designs for the trainers was also developed. The designs presented explore a broad range of architectural dimensions. Figure 7 summarizes the designs developed. The designs numbered 1 - 6 are the different system architectures, and the letters A - D indicate the different trainer architectures. The system designs 1-6 represent the entire SCS architecture. The trainer designs A-D are subset architecture designs for the SCS portion of all the trainers.

#	NAME	DESCRIPTION
1.	Monolithic Host	A single host for all SCS functions.
2.	Programmable Switch	A programmable switch connects hosts to trainers.
3.	Local Host Network	Local hosts connected via a network.
4.	Network Combined	Trainer hosts combined plus a network.
5.	Shared Host Network	Distributed network with shared hosts.
6.	Autonomous Trainers	One host per trainer, no network.
A.	DMS Kit	GFE DMS Kits are used.
B.	DMS Compatible	DMS components or DMS like components.
C.	PCTC based	DMS simulated in software on a host CPU.
D.	Distributed non-DMS	No DMS Kits, processors on a network.

Figure 7.   SCS Top Level Conceptual Designs

Figures 8-1 through 8-6 illustrate the six system designs. The following paragraphs discuss each of these top level designs.

The Monolithic Host (System Design 1) means that all SCS functions would be performed on one single host computer. All trainers and facilities would be connected to the single host computer via point to point connection methods. This design is simple, using a single CPU type and single operating system, and provides straight forward centralized control. Many successful computer systems have been built in the past using this architecture.

The Programmable Switch (System Design 2) provides multiple host computers that can be switched to support any trainer or any facility. This provides a high degree of fault tolerance and reconfigurability. Point to point connections are used.

The Distributed Network with Local Hosts (System Design 3) means that the trainers and facilities are connected via a network, but that each trainer and facility has one or more dedicated host computers directly connected to it. The network facilitates quick convenient communication between computers, aids in configuration management, and provides a means for centralized control. This design represents the current thinking in computer system design.

The Distributed Network with Combined Subsystems (System Design 4) is a variant of Design 3. The basic idea is that some of the trainers may be able to use the same local host computer, thus reducing the number of computers required.

The Distributed Network with Shared Hosts (System Design 5) means that none of the trainers have dedicated local hosts. The host computers are connected to a network and are only dedicated to a trainer for a particular training session. All data passed between the host and trainer for that session must now pass over the network. This design provides good fault tolerance and reconfigurability.

The Autonomous Trainers (System Design 6) means each trainer and facility has a dedicated host directly connected, and the computers are not connected. This design is simple, and would have a lower cost than ones that include a network. Many successful systems have been built in the past using this design.
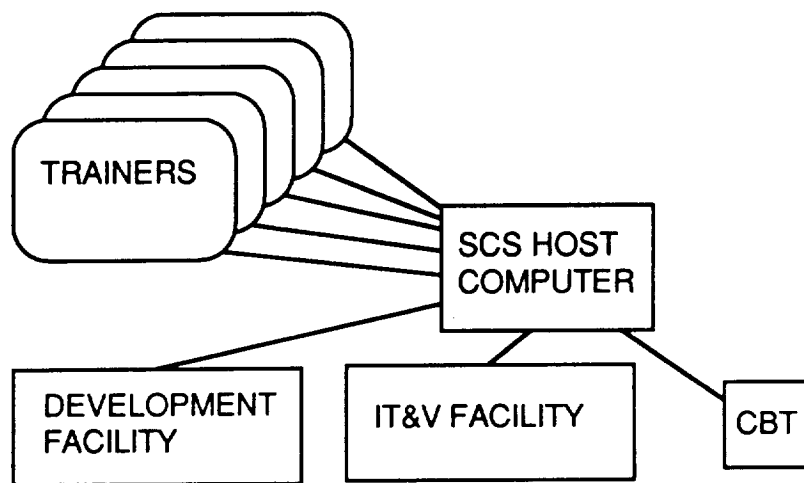
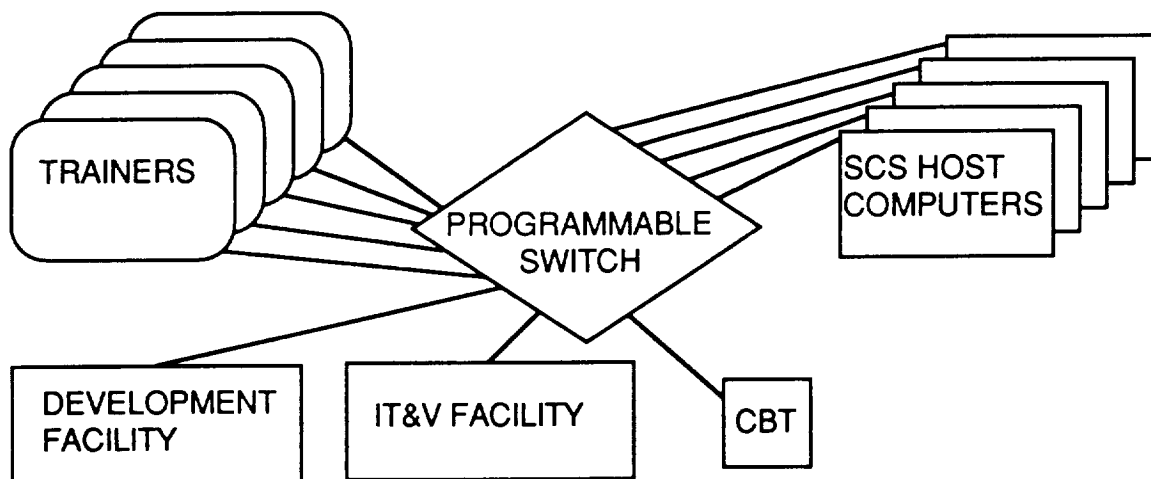Figure 8-1. Conceptual System Design 1 - Monolithic Host

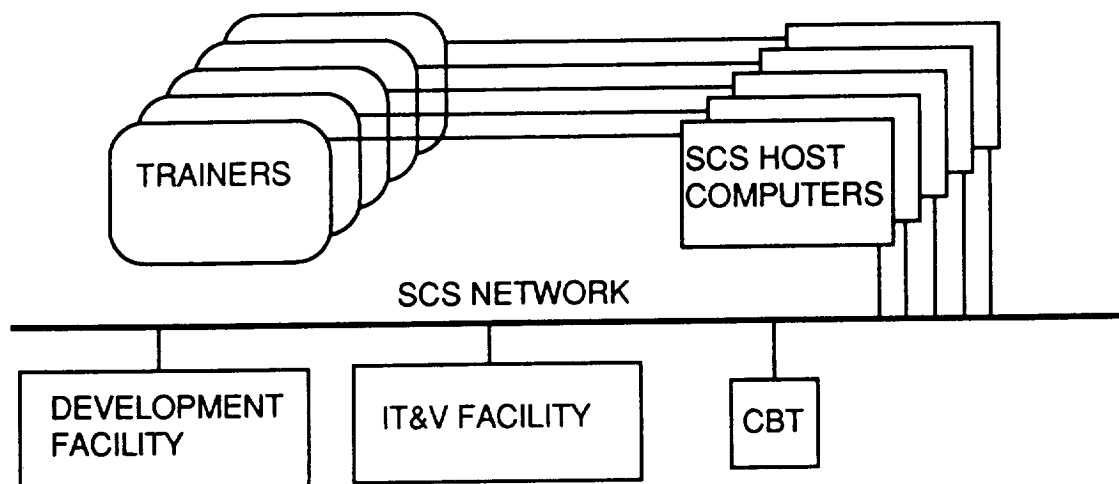Figure 8-2. Conceptual System Design 2 - Programmable Switch

Figure 8-3. Conceptual System Design 3
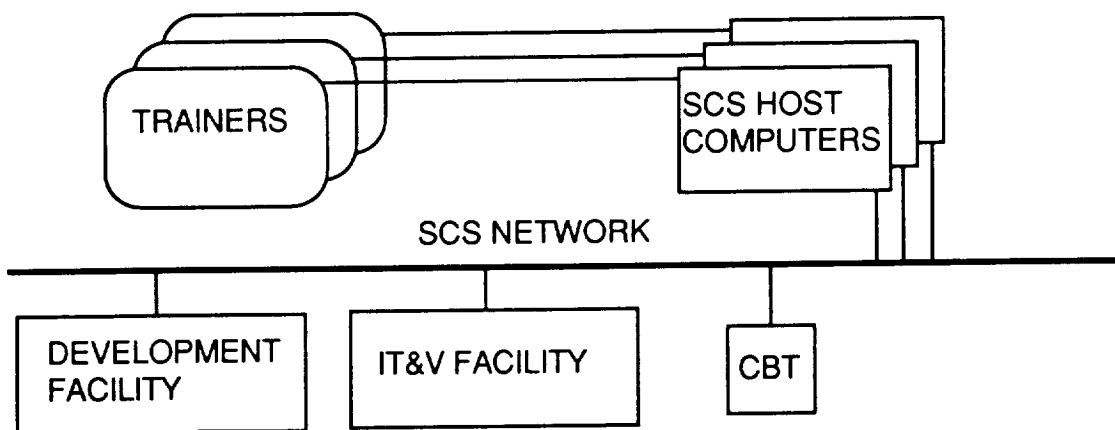- Distributed Network with Local Host

Figure 8-4. SCS Conceptual System 4 - Distributed
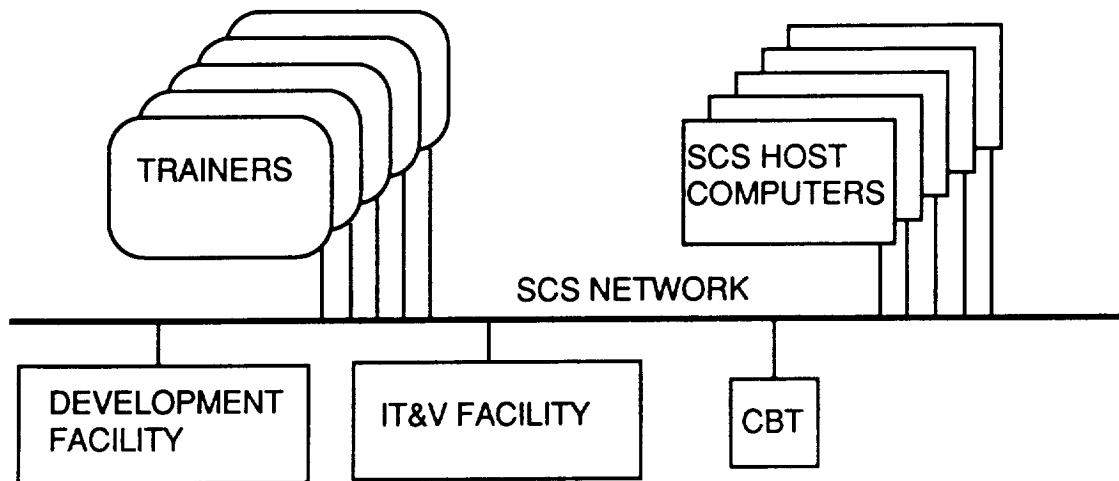Network with Combined Subsystems

Figure 8-5. SCS Conceptual System 5 -
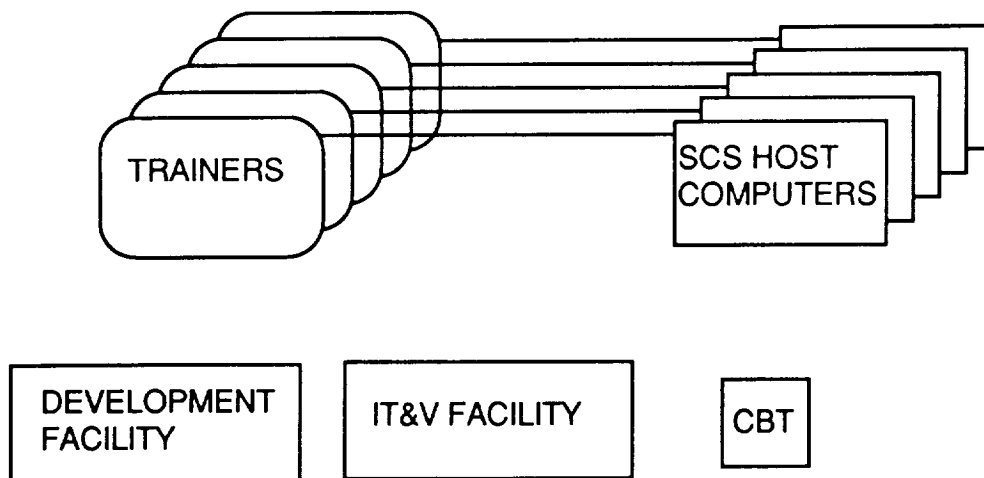Distributed Network with Shared Hosts

Figure 8-6. SCS Conceptual System 6 -
Autonomous Trainers, No Network

Figure 10 illustratesSCS trainer designs. Paragraphs following discuss these.

The DMS Kit trainer design (Trainer Design A) uses DMS hardware to support core system simulations and payload flight software/hardware. This would promote easy interfacing of SCS payload simulations to the SSTF, since the SSTF will use DMS Kits. It also means that SCS hardware would not have to support all of the core and payload flight simulations.

The DMS Compatible trainer design (Trainer Design B) uses partial DMS Kits or DMS kits without a SIB to support the core simulations needed for training. The lack of a SIB would make it more difficult to interface SCS simulations.

The PCTC-Based trainer design (Trainer Design C) means that the core systems would be all simulated in software on a host computer. No DMS or compatible hardware would be used. This would cut hardware costs, but increase software costs. Transportability to the SSTF would be adversely affected.

The Distributed Non-DMS trainer design (Trainer Design D) also means that core systems would be simulated in software, but that distributed processors would be used instead of a host computer. If COTS 80386 processors or personal Computers (PCs) could be used, some modified DMS software might be usable. This approach could provide the advantages of simulating core systems with hardware that is readily available and more easily maintained than the custom DMS Kit hardware.

By considering the six system level designs (1-6) in combination with the four trainer designs (A-D), 24 possible designs emerged. Using the evaluation criteria and the SCS requirements, six integrated design combinations were selected as the most viable candidates for refinement in SCS Study Task 5. The six selected are presented in Figure 9 below:

#	Top Level Designation	Name
I	3-A	Network Local Host - DMS Kit Trainers
II	3/5-A	Network Local Host and Shared Host - DMS Kit Trainers
III	4-B	Network Local Host with Combined Components DMS Compatible Trainers
IV	3-C/D	Network Local Host - PCTC Based/Non-DMS Trainers
V	2-A	Programmable Switch - DMS Kit Trainers
VI	3-A/D	Network Local Host - DMS Kit/Non-DMS Trainers

Figure 9.   Six Selected Designs
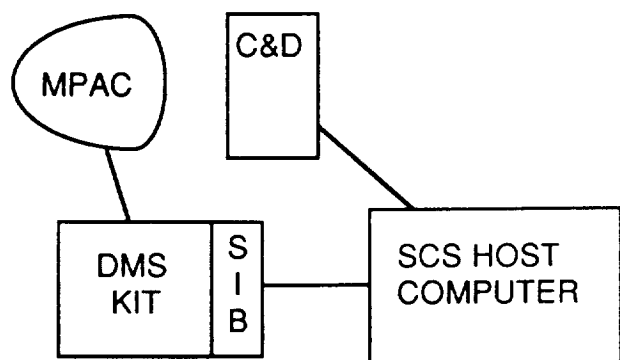
Figure 10-1. Trainer Design A
DMS Kit Approach

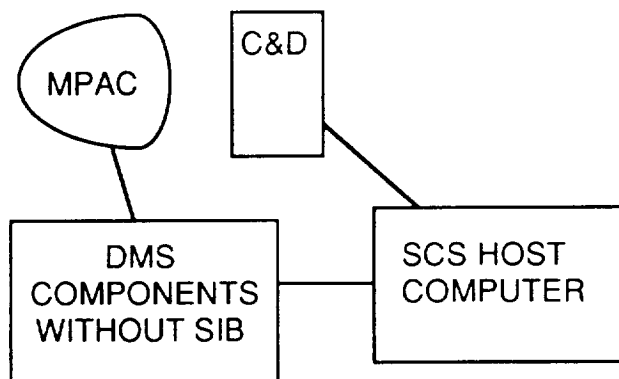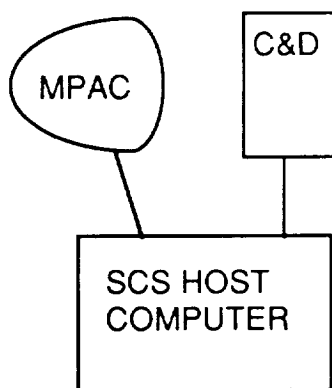Figure 10-2. Trainer Design B
DMS Compatible Approach

Figure 10-3. Trainer Design C
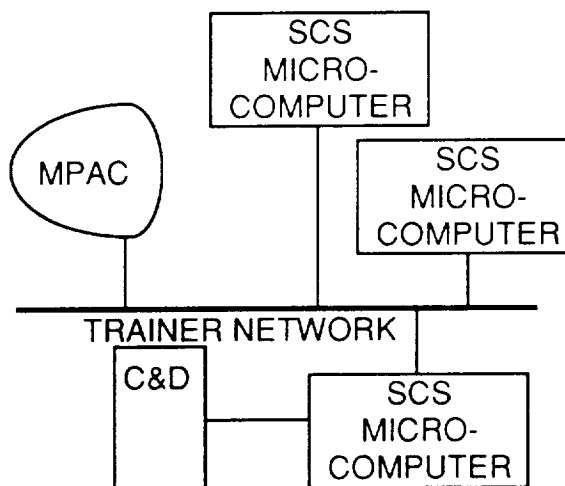PCTC Based Approach

Figure 10-4. Trainer Design D
Distributed Non-DMS Approach

Figure 10. Trainer Designs

Note: These trainer designs illustrate one set of connection choices. Other choices are supported by the SCS designs. Details of other supported choices are discussed in the Refined Conceptual Design Report.

The Top Level Designators (see Figure 7) were very useful for reference during the many discussions of the numerous designs reviewed in the process of selecting these six. Each design under consideration was thus tied back to the very top level concepts, and this helped ensure the consideration of the broadest possible range of designs. Pictures of the selected six designs are shown in Figures 11-1 through 11-6.

A paragraph describing each of the six follows.

I.   **SCS Integrated Conceptual Design 3-A**
       Integrates the Distributed Network Local Host architecture with the DMS Kit trainer design. The local hosts in each of the trainers perform all real-time simulation activities required to support payload training.

       The distributed network allows maximum flexibility for high speed communications between the SCS facilities or subsystems. Any facility can exchange data with any other facility using a single interface. The implementation of a single local trainer host for payload simulation executive functions is less complex from a system software viewpoint than implementing shared hosts at the SCS level.

       The use of the DMS Kits, including the SIB, is the SSE recommended approach to Space Station system development, integration, testing, and training. It is also the approach favored by the SSTF development effort at JSC. The use of the DMS Kit helps guarantee a high level of fidelity for payload training. Flight equivalent Space Station systems and payloads are easily integrated into the trainer with the DMS Kit. Also, Core system functional simulations, software developed for the SSTF, and SSE developed software would be directly transportable to the SCS. Likewise, PTC developed experiment models would be more easily transportable to the SSTF if developed in a DMS Kit environment. The SIB offers a great deal of functionality useful for simulations. The SIB simplifies the implementation of some training requirements, like fault insertion.

II.   **SCS Integrated Conceptual Design 3/5-A**
       This design integrates a combination of the Distributed Network Local Hosts and Distributed Network Shared Hosts architecture with the DMS Kit trainer design. Individual shared hosts are allocated to perform real-time or non real-time functions, but not both. The real-time shared hosts support a different training scenario for each trainer. The local hosts support real-time training functions specific to a particular trainer.

       This design is somewhat similar to 3-A (# I) above and shares many of the advantages discussed. The use of shared hosts for non real-time functions offers additional flexibility, increased fault tolerance, and allows more powerful, more cost effective hosts to be utilized than a design with only dedicated hosts.
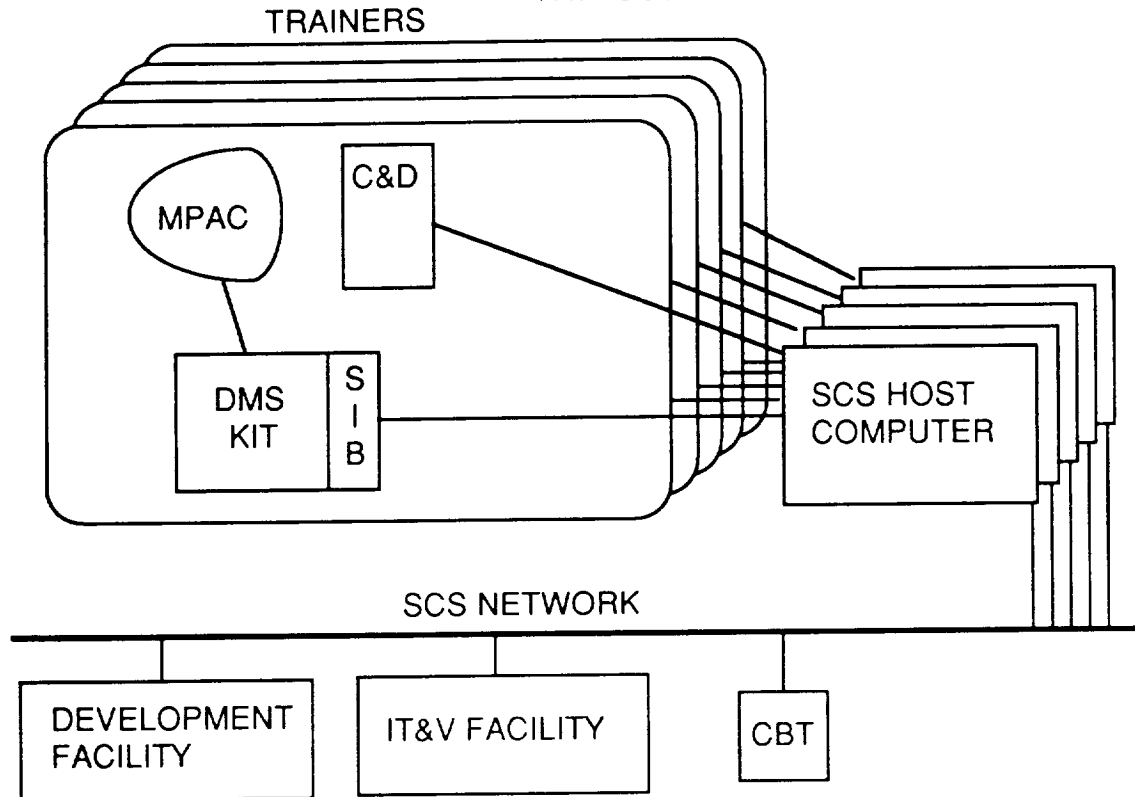
TRAINERS

Figure 11-1. Integrated Design I : Network Local Host - DMS Kit (3-A)
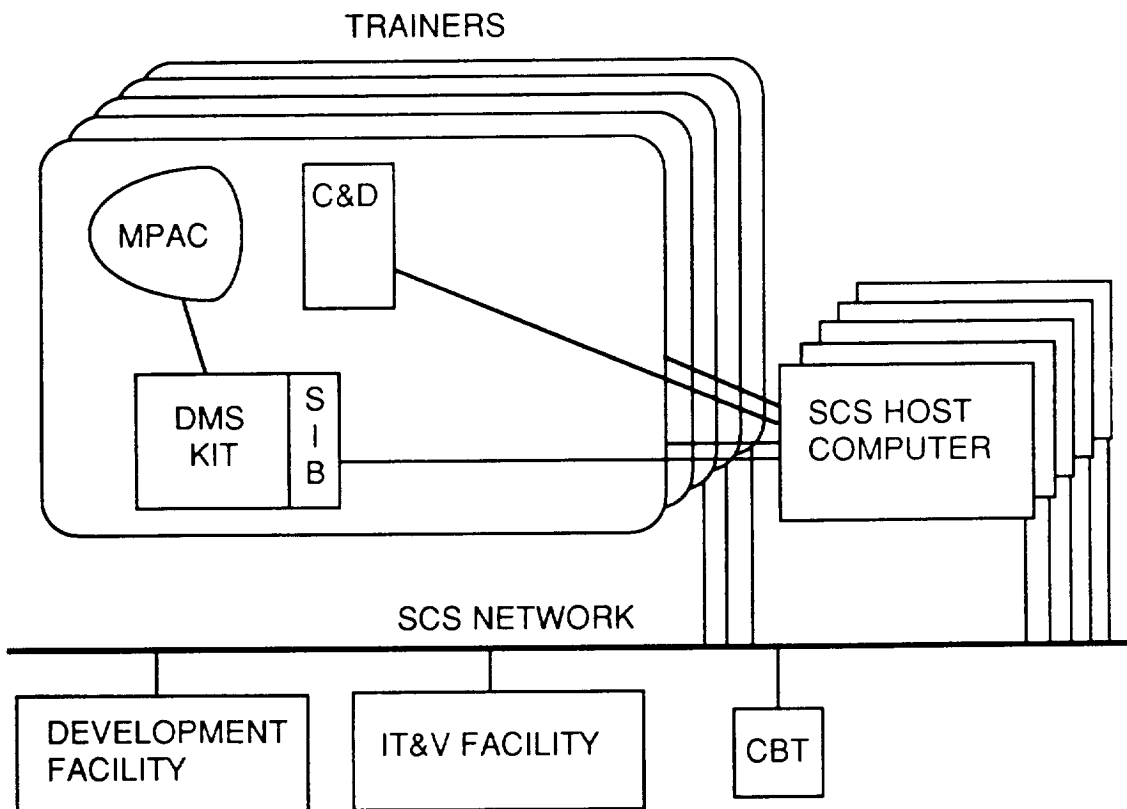
TRAINERS

Figure 11-2. Integrated Design II : Network Local Host/Shared Host - DMS Kit (3/5-A)

III. <u>SCS Integrated Conceptual Design 4-B</u>
Integrates the Distributed Network with Local Hosts and Combined Training Components architecture with the DMS Compatible trainer design. The Combined trainer is integrated with the Attached Payload trainer, and other trainers might be combined.
This design utilizes a distributed network, like 3-A (# I) above. By combining trainers, a savings in equipment cost and possibly facility space can be obtained. The Part Task Trainers could also be combined such that a single host could support multiple Part Task Trainers.

IV. <u>SCS Integrated Conceptual Design 3-C/D</u>
Integrates the Distributed Network Local Host architecture with the synthesis of the PCTC-based trainer and the Distributed Non-DMS trainer. This architecture does not have a DMS Kit or DMS components. Some trainer functions are implemented on the trainer host and some functions are implemented on dedicated processors.

This design also utilizes a distributed network. The trainer design, with no DMS hardware components, offers flexibility of hardware configuration, and reduces risk resulting from uncertainties in the DMS Kit development schedule. In addition, COTS non-DMS hardware can be readily purchased from a vendor and is certain to be less expensive than DMS hardware. The use of non-DMS hardware does not necessarily preclude the use of DMS software. It is likely, however, that DMS software would require some degree of modification to run in a non-DMS hardware environment.

The use of PCTC-based trainers would give the economic advantage of starting from an existing facility which could evolve into the finally required trainers. The advantage of using both the PCTC-based trainers and non-DMS trainers is that neither of these trainer designs contains DMS components, and the opportunity for synthesis between these two types of trainers thus seems good. The non-DMS design is a distributed, microcomputer based design that should compliment and perhaps off set some of the disadvantages of the somewhat monolithic PCTC-based design.

V. <u>SCS Integrated Conceptual Design 2-A</u>
Integrates the Programmable Switch with multiple host architecture with the DMS Kit trainer design. In this design, multiple trainers may be interfaced to the same host. The trainers may be switched and reconfigured quickly in the event of a host failure.

The use of shared hosts makes maximum use of system resources. Any trainer can be quickly configured with any host, providing increased flexibility and fault tolerance. The use of dedicated point to point interfaces between the trainers and the hosts ensures that communication bandwidth problems are minimized.
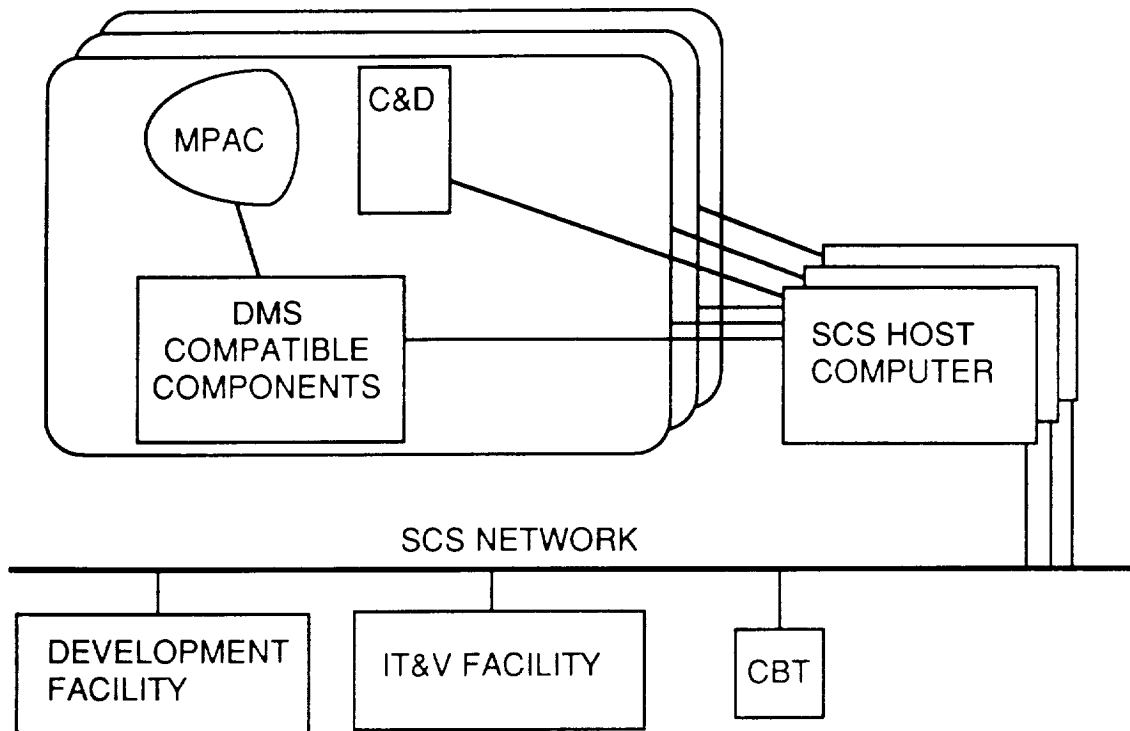
TRAINERS

Figure 11-3. Integrated Design III : Network Local Host with Combined
Componenets - DMS Compatible (4-B)

TRAINERS
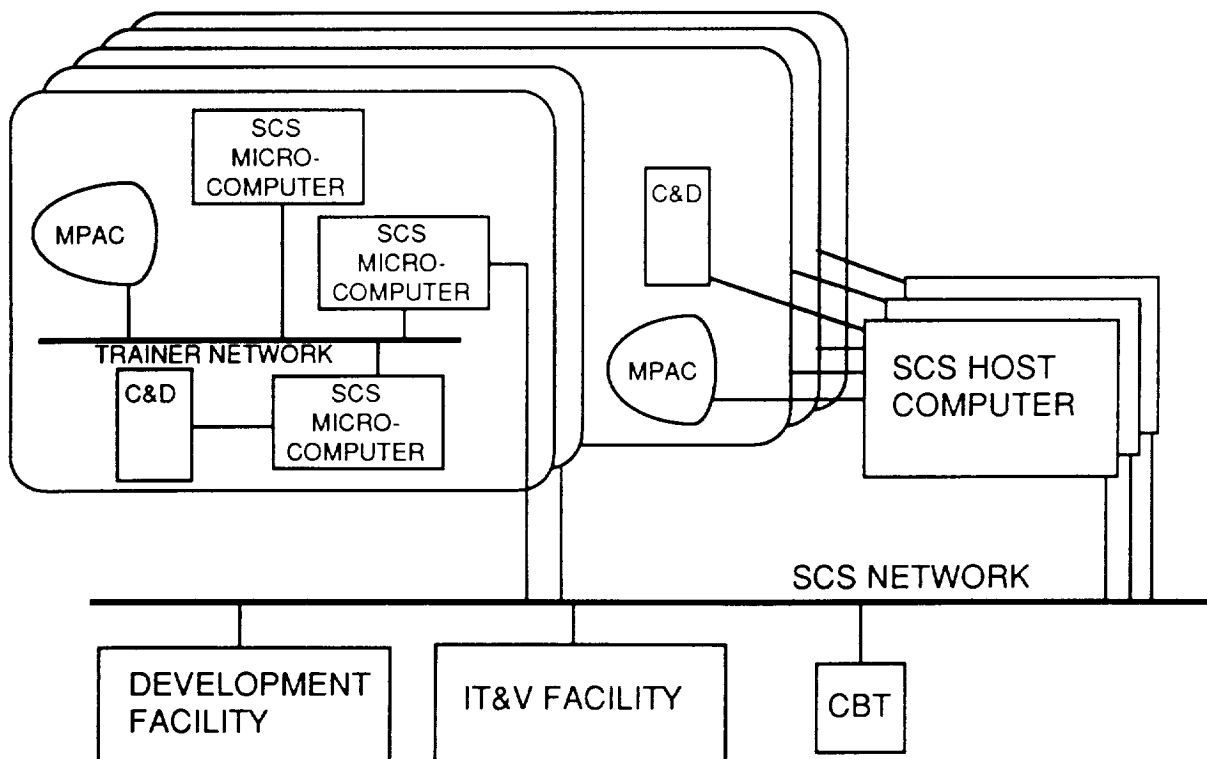
Figure 11-4. Integrated Design IV : Network Local Host -PCTC
Based/Non-DMS (3-C/D)

This design has a lot of promise. However, investigations conducted at the beginning of the Refined Design Task revealed that a switch capable of switching the required wideband high rate Direct Memory Access (DMA) channels currently does not exist. Existing switches can only handle 8 bit wide low rate channels.

VI.   <u>SCS Integrated Conceptual Design 3-A/D</u>
Integrates the Distributed Network Local Host architecture with a combination of the DMS Kit trainer and the Distributed Non-DMS trainer. In this design, Non-DMS trainer elements are integrated with DMS trainer elements. Non-DMS trainer elements such as generic processors and peripheral devices are directly connected to the DMS LAN, instead of being directly connected to the SIB. In addition, elements of the Combined system approach are implemented in that the trainer host and SIB are shared across multiple trainers.

This design is similar to 3-A (# I) above. The use of some non-DMS components in the trainer provides additional flexibility, and allows increased trainer functionality. Functional areas where non-DMS components could be desirable are instructor control and monitoring, audio/video systems, Core systems interface, and payload simulation control. These areas are envisioned to be implemented on the trainer host in other designs, but there could be advantages to implementing these functions in a processor directly attached to the payload LAN.
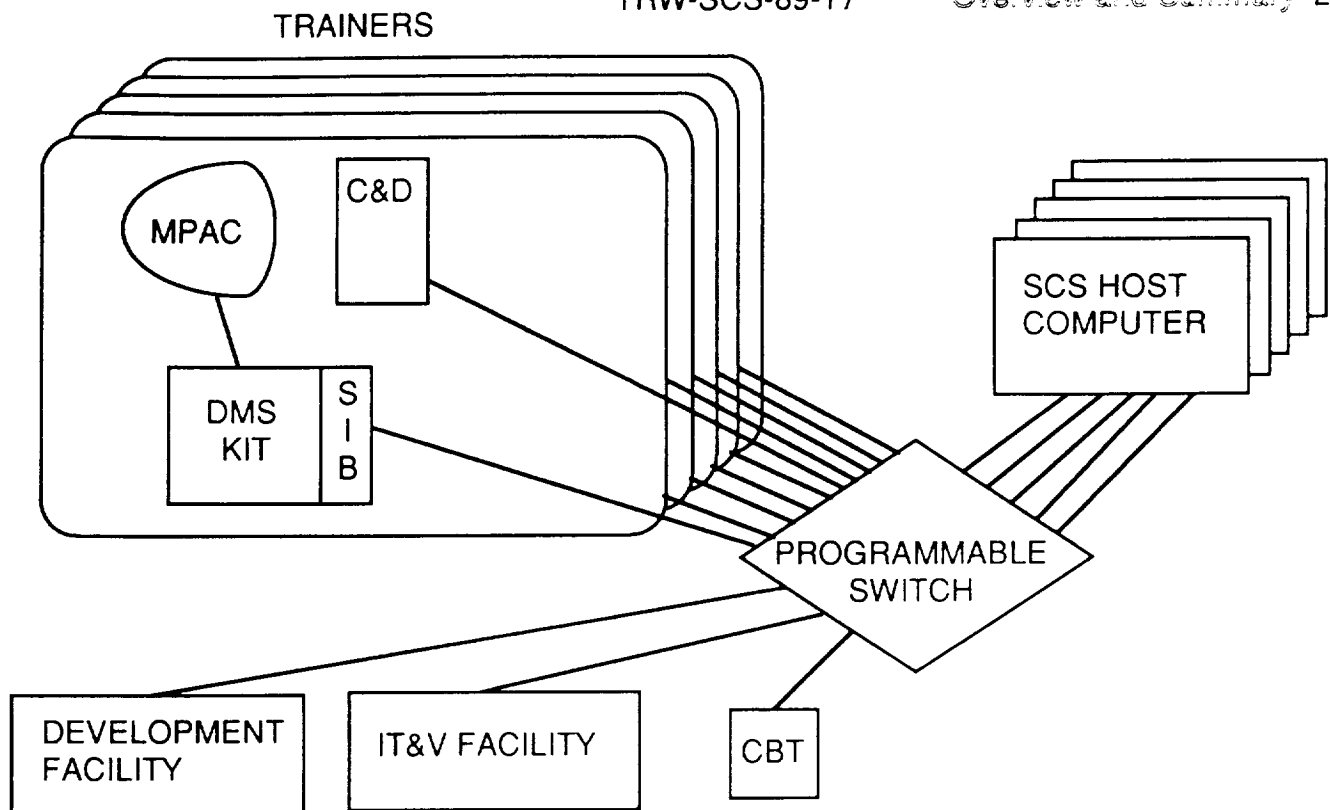
TRAINERS

Figure 11-5. Integrated Design V : Programmable Switch - DMS Kits (2-A)
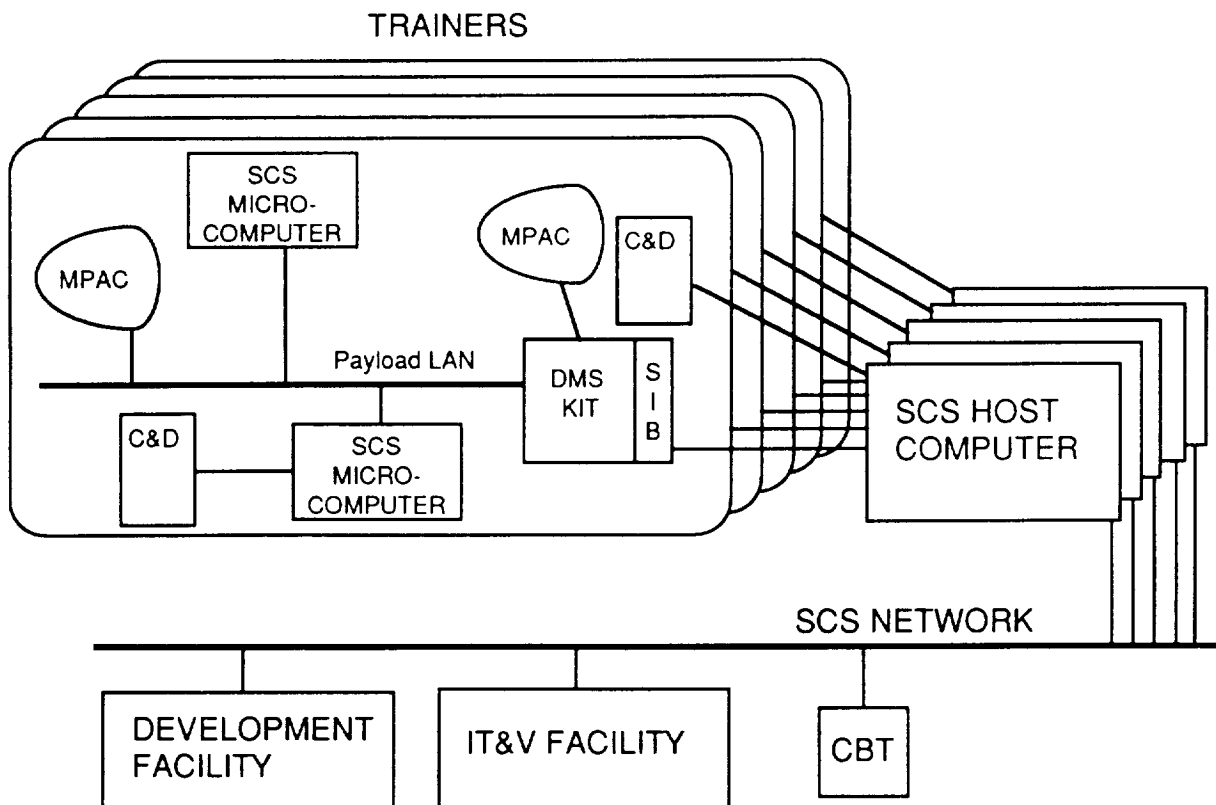
TRAINERS

Figure 11-6. Integrated Design VI : Network Local Host - DMS Kit/Non-DMS (3-A/D)

Task 5 - Refine SCS Conceptual Designs

The first step of Task 5 was to select the three designs to be refined. This involved review of the current SSFP DMS Kit, SIB, and SSE design materials and presentations. Lockheed was contacted directly to ascertain the current design of the SIB, a very key simulation element. It became clear from this effort that the DMS Kit and SIB designs are far from settled. This information obviously affected the choice of the three designs to be refined. The result was that the designs selected for refinement are broader and cover a wide number of options.

Once the three designs were settled on, lists of the possible choices of the design details common to all SCS designs were made. These proved very useful in refining the three designs selected. Due to the uncertainties of the basic DMS Kit design - e.g. the method of connecting the SIB to the simulation host has not been finalized - the lists of choices of the details are quite extensive. Many options for C&D panel connections, payload representation, and flight equivalent payload connection were considered.

Design work on the three then proceeded. Choices of details were made based on the best information available, the years of experience of the members of the contractor team in systems design, and projections of where various technologies will be when the actual design process really will begin. The beginning of the SCS design effort is currently estimated to be two years from the time of this report.

The three designs were filled out to a lower level of detail, and analysis of the loading on each design was performed. Design of the elements that proved to be the same (the development facility, the IT&V facility, CBT, and POIC trainer) in all three designs was also done. Various payload software simulation and payload flight equivalent interface options were investigated.

Finally, a comparison of the three selected designs was completed based on all the analysis performed as part of Task 5. The details of the results of Task 5 are contained in the final report Volume 3 "Detailed Conceptual Design Report".

TASK 5 RESULTS SUMMARY

The three designs selected and refined are the key result of Task 5. The three selected designs map to the six recommended in the Task 4 effort, but are not simply three of the six recommended. Due to the uncertainties in DMS Kit and SIB design uncovered in the first part of the Task 5 effort, it became clear that the three Task 5 designs selected to be refined needed to be broader than originally envisioned. Discussions with all members of the SCS team, including NASA, yielded the fact that the Task 5 investigation must address three possibilities:

1) The DMS Kits and the SIB will be available when needed by SCS, and the SIB will only connect to a host computer through a point to point parallel connection (like a Digital Equipment Corporation VAXBI 32 bit wide parallel interface).

2) The DMS Kits and the SIB will be available when needed by SCS, and the SIB will connect to a network through a high speed fiber optic connection that will connect to a network.

3) The DMS Kits and the SIB will not be available when needed by SCS, or if available, are designed to meet flight system development and not training functions.

The designs selected to help address the above three possibilities and be refined were given names in order to avoid any indication at the outset of Task 5 that one design was favored over the others. The three selected are:

LOCAL HOST - This is the Network Local with DMS Kit Trainers (top level designation 3-A) and Task 4 recommended Design I (3-A : Network Local Host - DMS Kit). It addresses possibility number 1 above.

SHARED HOST - This is the Shared Host with DMS Kit Trainers (top level designation 5-A) and a variant of Task 4 recommended Design II (3/5-A : Network Local Host/Shared Host - DMS Kit). It addresses possibility number 2 above.

DMS EQUIVALENT - This is the Network Local Hosts with Non-DMS Trainers (top level designation 3-D) and a variant of Task 4 recommended Design IV (3-C/D : Network Local Host - PCTC Based/Non-DMS). It addresses possibility number 3 above. The PCTC Based option, although completely viable, was not included for further study because there is little need to explore through a "what if" design study a facility and design that currently exists. Also, this approach represents design methods reflecting state-of-the-art design from when the PCTC was developed, not current system design thinking.

A legend is provided as Figure 12 to designate individual components of each design. A brief discussion of each design is given in the following paragraphs.

LOCAL HOST

In the Local Host design, a separate host computer is dedicated to and directly connected to each trainer and facility. The SCS Network facilitates communication of mostly non real-time information (e.g. software loads, CM data, courseware, and data files of all types) between host computers. Some real-time data would pass over the SCS network since the instructor stations are shared among trainers via the SCS. Figure 13-1 presents an overview of the Local Host designs.

Four of the part task trainers have DMS Kits, and five do not. This is primarily an economy measure based on the projected costs of the DMS Kits and the acceptability of some lower fidelity training. The non-DMS Kit trainers will be of lower fidelity and lesser capability for integrating flight equivalent hardware and software than the DMS Kit based part task trainers.

POIC Workstations

Developer Workstations

Disk

Instructor Stations

Module

Nodes

CBT Stations
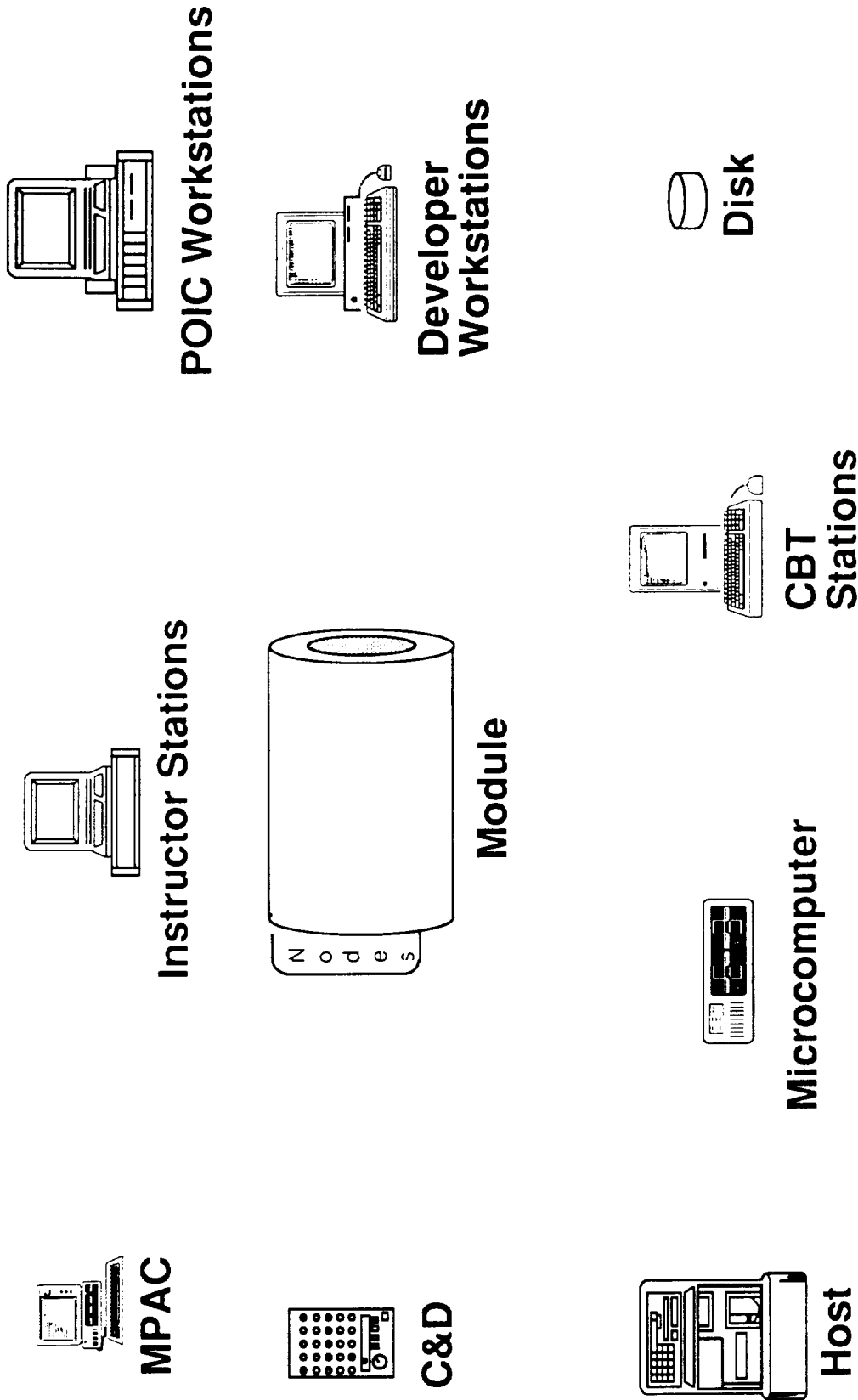
Microcomputer

MPAC

C&D

Host

Figure 12. Legend for Top Level Designs

Note that the three designs show selected options for C&D panel connections, payload representation, and core system options. Other supported options are discussed in the Refined Design Report.
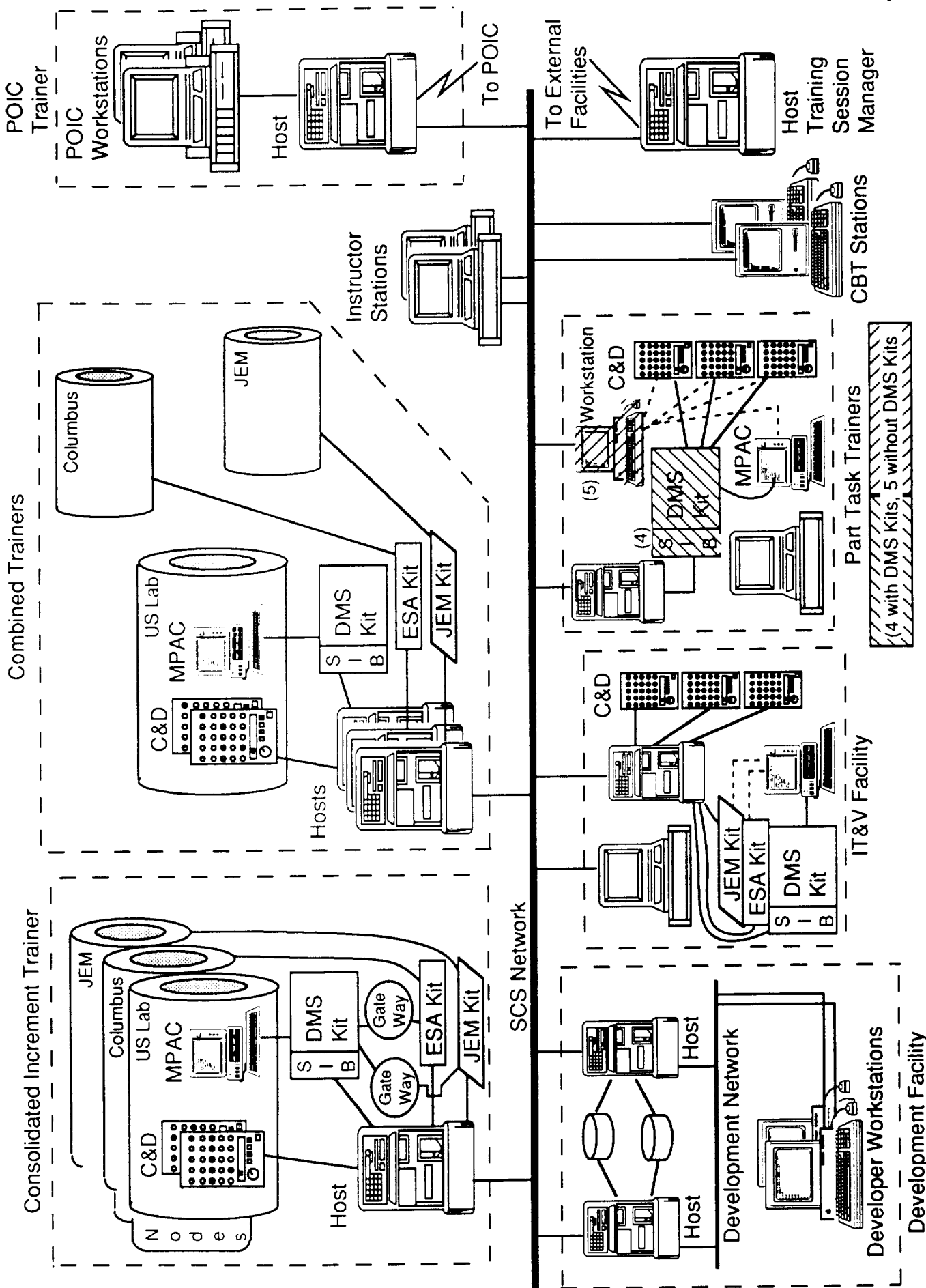
Figure 13-1. SCS Local Host Design

9/26/89

The POIC host communicates directly to the POIC to facilitate training sessions with the POIC. The SCS executive software, called the Training Session Manager, resides on a separate host. This software controls set up of the individual training sessions, and communicates to other external facilities (shown in Figure 4 previously). Thus, for example, mission plans would come from the MPS to the training session manager host, and then go as needed via the SCS network to the various local host computers.

The Local Host design is formulated to:

- use DMS Kits and other SSF compatible components
- accept flight equivalent payload hardware and software without significant modification
- isolate and minimize the real-time traffic loading on the SCS LAN
- sustain a level of fault tolerance
- interface directly with SSF support systems, development systems, and communications systems

SHARED HOST

In the Shared Host Design, host computers serve trainers via the SCS network. The hosts are dedicated to a particular trainer only for a particular training session. This means that a great deal of real-time traffic must pass over the SCS Network. However, this sharing of host greatly increases fault tolerance, flexibility, and reconfigurability. Figure 13-2 presents an overview of the Shared Host design.

For the non-DMS Kit part task trainers, a high end workstation was projected to be the best design, rather than more hosts on the network. Workstations are currently fairly powerful and inexpensive and will be even more powerful and less expensive in the coming years. In this design, workstations for part task trainers proved to be a better choice than a larger more complex network.

The other features of this design bear many similarities to the Local Host design in terms of the overall structure and in details other than those already mentioned.

The Shared Host Design is formulated to:

- be highly distributed
- be fault tolerant
- be reconfigurable
- use DMS Kit and other SSF compatible components
- accept flight equivalent payload hardware and software without significant modification
- interface directly with SSF support systems, development systems, and communications systems
- maximize SCS LAN usage
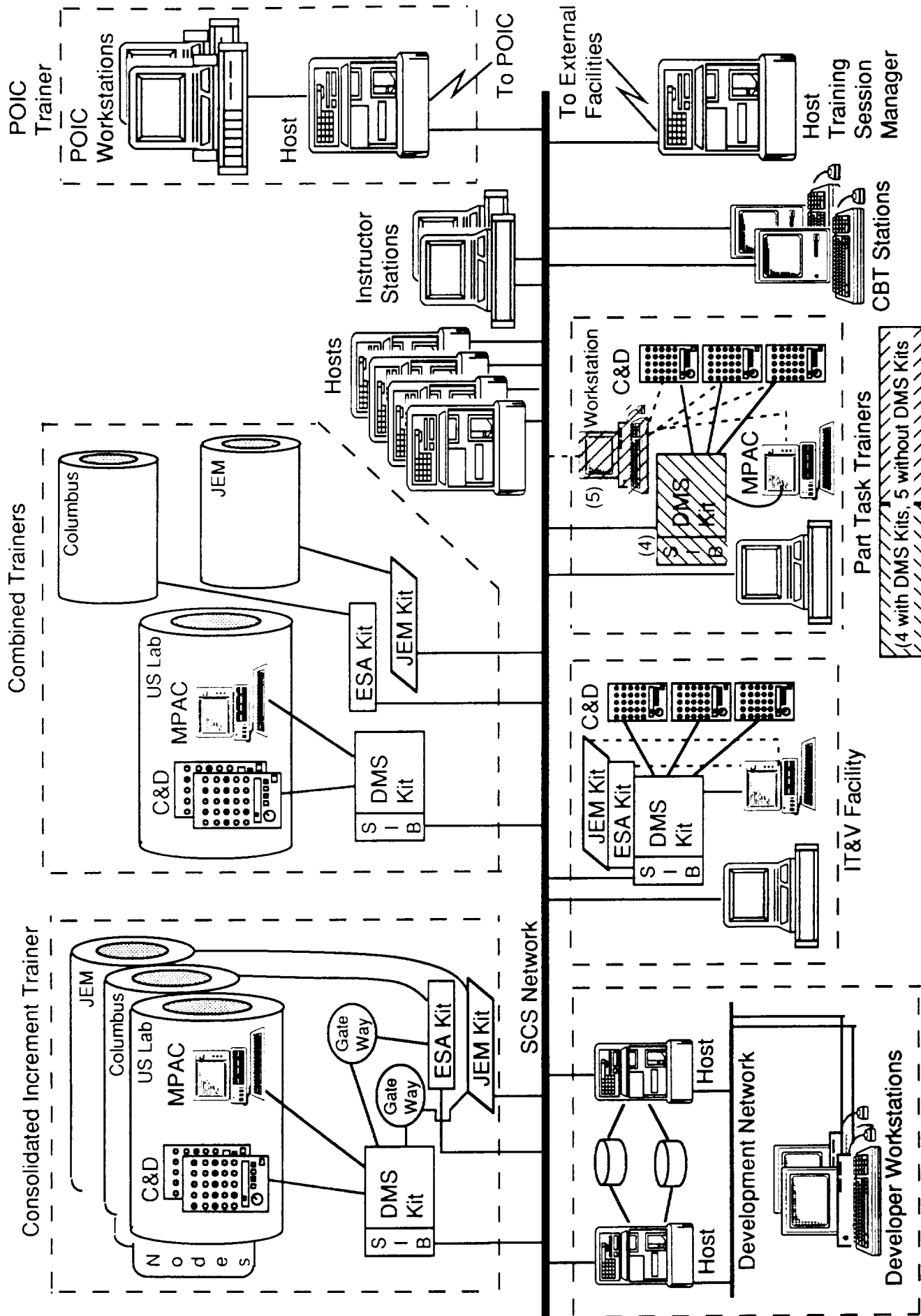
Figure 13-2. SCS Shared Host Design

9/26/89

DMS EQUIVALENT

In the DMS Equivalent design, the main feature is the absence of DMS Kits. The approach used is to replace the functionality of the DMS Kits with standard commercial off-the-shelf (COTS) hardware. Figure 13-3 presents an overview of the DMS Equivalent design.

The overall DMS Equivalent Design is very similar to the Local Host Design in almost all features. The difference is that the DMS Equivalent Design has dedicated processors in place of both the local hosts and the DMS Kits. Given the increasing power and decreasing cost of processors (either single board computers or micro computers), this approach has high appeal, especially given the current projected two years before a full up SCS design effort begins.

The DMS Equivalent Design is formulated to:

- implement SCS requirements without the use of DMS Kits
- be a highly distributed system
- minimize SCS network traffic loading
- use the same overall architecture as the Local Host Design
- use off-the-shelf equipment
- be able to adapt certain DMS software
- be fault tolerant

Design Comparisons

The three SCS designs share many similarities in spite of their different architectures. Figure 14 is a summary of the key design differences in the areas that are important designs aspects. Figure 15 is a summary of the results of the comparison effort done as part of the SCS Study Refined Design (Task 5). The factors are those that were derived at the beginning of the Conceptual Design (Task 4) portion of the SCS Study. Further details and more detailed comparisons are contained in the "Refined Design Report".
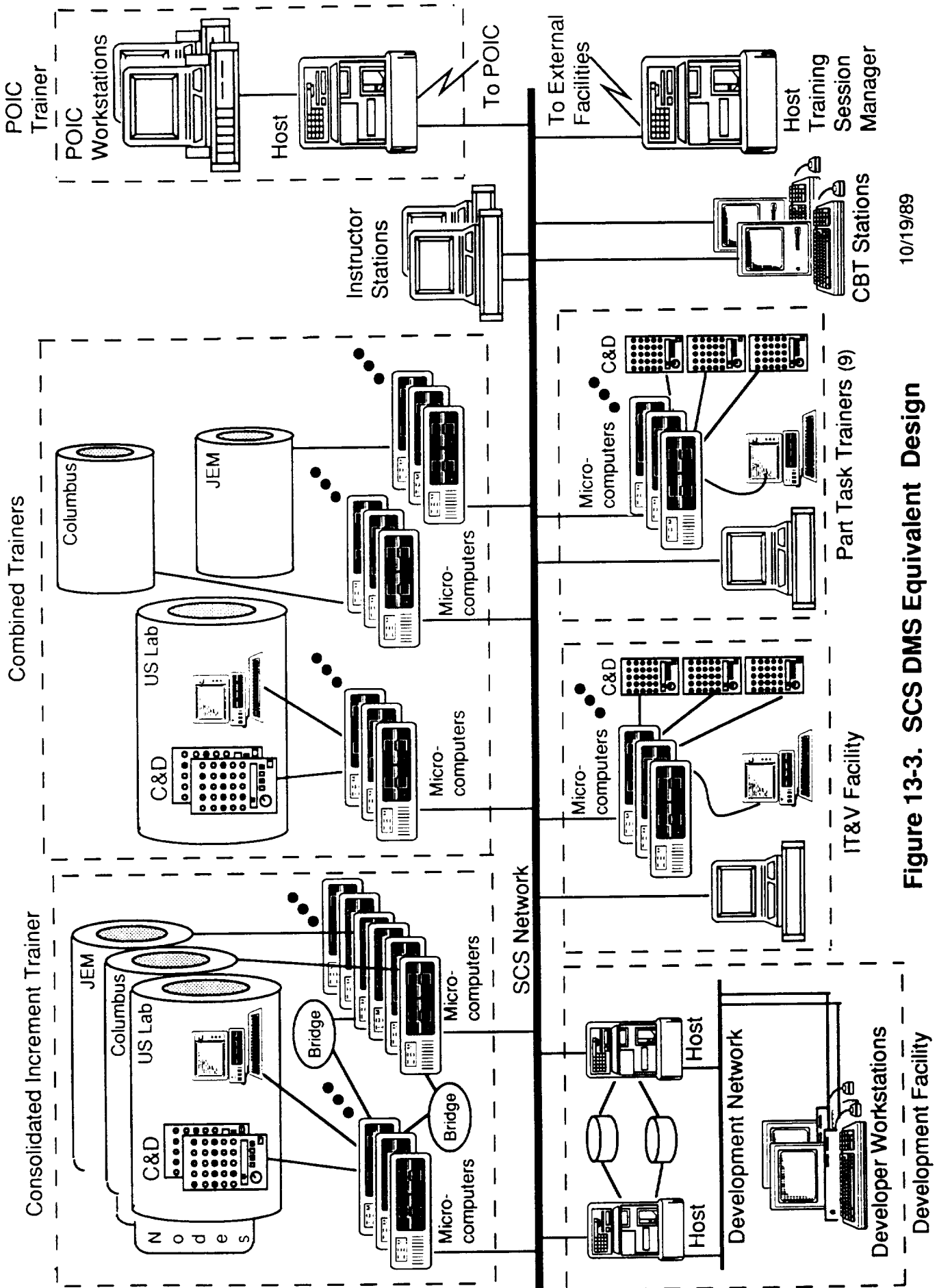
Figure 13-3. SCS DMS Equivalent Design

10/19/89

(Multiple microcomputers replace both host computers and DMS/ESA/JEM Kits within the trainers)

Design Aspects	SCS DESIGNS		
	Local Host	Shared Host	DMS Equivalent
Trainer Host	Single Dedicated Host	Multiple Hosts	Multiple Microcomputers Distributed On Dedicated Trainer Lan
Trainer Interface	SCS LAN to Host	Host SCS LAN to SIB	SCS LAN to Microcomputer
Flight Equivalent Payload Interface	DMS MDM	DMS MDM	I/O Adapter on Processor bus
C & D Panel Interface Chosen	Direct to Host	DMS Local Bus	Direct to Microcomputer
Payload Models	In Trainer Host	In Trainer Host	In Dedicated Microcomputer(s)
C & T Model	In Dedicated C&T Controller	In Dedicated C&T Controller	In Dedicated Microcomputers
High Rate Link	From Trainer Host to C&T Controller	From DMS SDP to C&T Controller	Dedicated Microcomputer on LAN
Audio /Video System	Adapter / Controller on Host bus	A/V Processor /Controller on SCS Lan	Adapter / Controller on Dedicated Microcomputer
Part Task Trainer	Two Types: DMS-Based with Dedicated Host — — — — — Non DMS Workstation Based	Two Types: DMS-Based sharing SIB and Multiple Host on SCS Lan — — — — — Non DMS Workstation Based	One Type: DMS Equivalent based with Multiple Microcomputers on a dedicated Trainer LAN
Type of Data on SCS Network	Mostly Non-realtime development data, except real-time instructor commands	Real-time model to DMS data + instructor commands + non-real time development data	Mostly Non-realtime development data, except real-time instructor commands

Figure 14. Summary of Design Differences

FACTOR	LOCAL HOST	SHARED HOST	DMS EQUIVALENT
Reliability/ Maintainability	Good	Good	Good
Expandability/ Scalability	Limited	Good	Good
Cost	High	Lower	Lowest
Computing Headroom	Limited	V. Good	Good
H/W & S/W Standards	Good	Good	Flexible
Reconfigurability/ Modularity	Limited	Good	V. Good
Ease of Operation	Good	Good	Good
Performance vs Cost	Fair	Better	Best

Figure 15.   Summary Results from a Comparison of SCS Designs

Task 6 - Develop and Present Technology Demonstrations

A list of technology demonstrations that could be presented to help reduce the SCS development risk was developed as the first step of this task. An investigation including phone calls and literature searches was then made into each of the areas on the list to help determine relevance and importance to SCS development and operation. Next, based on this investigation, the list was prioritized into three categories:

1.) Highly Relevant
2.) Very Relevant
3.) Relevant

The list was then reviewed with NASA to determine which of the demonstrations would be the most important and beneficial.

Following is the list of demonstrations that were prioritized as "highly relevant":

1.) Design Based -- Demonstrate a useful concept based on the designs proposed by the SCS study.

2.) Switches -- Demonstration of what type of switching mechanisms may be utilized by the SCS.

3.) SSE Simulation Control Architecture -- Presentation of some very important and relevant SSE concepts and how they might affect SCS.

4.) Networks -- Presentation of types of networking that may be utilized in the SCS.

5.) UIMS (Dataviews and Labviews) -- Possible use of User Interface Management System (UIMS) tools in the SCS.

6.) Ada Technology -- Presentation by an expert in the utilization of Ada to review the advantages and risk of using Ada to develop the SCS.

Following is the list of demonstrations that were prioritized as "very relevant":

1.) Workstations -- Demonstrations of various workstations, their capabilities, and their relevance to the SCS problem.

2.) Video Generation -- Demonstrate what is available and relevant for SCS use.

3.) Generic/Programming Panels -- Demonstrate the feasibility of using these for "virtual panels", and the current state of the art of currently available hardware.

4.) Rapid Prototyping -- Demonstrate tools available, and their relevance to SCS.

5.) Simulation Language -- SimScript II.5 or languages like this could be demonstrated.

The list of demonstrations that were prioritized as "relevant":

1.) Computer Based Training -- Demonstrations disks were received from two vendors that can produce and execute courseware on a personal computer.

2.) Artificial Intelligence --Investigate this family of technology, and its current state. Applicability to SCS seems low at this time.

3.) Device Adapters -- Investigate the many options available in this area.

After discussions with NASA, the demonstrations to be presented were selected. As the SCS Study continued, a couple that had initially been eliminated rose in importance and were also given. The final list of demonstrations given, in the order they were given is:

• SSE Simulation Control Architecture -- Presentation of some very important and relevant SSE concepts and how they might affect SCS.

• Ada Technology -- Presentation by an expert in the utilization of Ada to review the advantages and risk of using Ada to develop the SCS.

• Design Based -- Demonstrate off-nominal training using flight software.

• Generic/Programming Panels -- Demonstrate the feasibility of using these for "virtual panels", and the current state of the art of currently available hardware. Also demonstrated Rapid Prototyping as part of this demonstration. Demonstrated tools available, and their relevance to SCS.

TASK 6 RESULTS

A summary of the important points and conclusions of each of the technology demonstrations follows. Where appropriate, further details of the demonstration are given to make clear the results.

SSE Simulation Control Architecture

If the SSE simulation architecture is used at all the centers developing SSFP simulations, this will help ensure that simulations developed at one center are transportable to other centers that require the same type of simulations. The following points summarize the current SSE simulation architecture:

- The basis of the SSE design is cyclic models. Demand/response models are being considered.

- Ada tasks will be used as the entities (collection of functions or routines) that can be scheduled for execution.

- Models are bound to software Simulation Harnesses (bind individual models together for execution) at compile time.

- A simple linked-list concept will be used for model execution.

- The SSE baseline design demands that software simulations run on one host computer.with some number "n" CPUs. A simulation harness task will be required for each model execution rate for each CPU.

- SSE will provide a generic procedure to surround each model component, thereby standardizing interfaces of models.

Ada Technology

Ada is planned to be the language used for virtually all SSFP software. Using a single language across a project has obvious advantages in terms of lower life cycle cost for software maintenance and software updates. In addition to committing to use Ada on the SSFP, NASA is considering the adoption of Ada as its standard language. Production quality Ada compilers are available, and Ada Computer Aided Software Engineering (CASE) tools are beginning to appear.

This presentation gave an overview of the Ada language, discussed software development and productivity issues. Software development methodologies and how they relate to Ada was also discussed. The advantages of the Object Oriented Design (OOD) software design methodology over the more traditional structured analysis (SA) method was discussed. In OOD, identifying objects, determining their attributes, and defining how the objects interact are the first steps of a software design effort. In SA, hierarchical entities are defined, and the actions and the order of the actions they must accomplish are the first steps of software design. A list of conclusions or caveats provide an excellent summary of the points made. These are:

- Do not expect big increases in productivity when you first use Ada.

- Do not expect software engineers and programmers to learn Ada in their spare time. Extensive Ada training is required to make good use of the language.

- Do not assume that a validated Ada compiler (one that is on the DOD approved list) is a good, production quality compiler.

- Do not expect traditional schedules and effort distributions to work on an Ada project. More time will be spent in design, less on integration.

• Do not expect traditional software development methodologies to work well for Ada and support Object Oriented Design (OOD).

• Beware of Ada tasking performance and scheduling problems. This specific of the Ada language has caused other projects problems since Ada tasks do not necessarily execute sequentially like FORTRAN.

• Plan, as much as possible, for other Ada-specific risks such as lack of personnel with Ada experience, choice of unsuitable Ada compilers and tools, lack of Ada project management skills, lack of experience estimating hardware and memory resources required by Ada, and choice of the wrong software development methodology.

Design-based Demonstration

The Spacelab Software Development Facility (SDF) was utilized to provide insight into the issues related to the use of a flight software based training facility. This demonstration approach was considered to be relevant because a flight software based SCS is a leading design candidate and the current JSC/SSTF design utilizes flight software and supporting simulations as the cornerstone of their trainer development. The demo encompassed three areas of interest to SCS:

a.) Off-nominal training in a flight software environment for a specific experiment.

b.) Validation automation (VALAUTO) for trainee absent mode, automated testing, and operations evaluation functions.

c.) Generic malfunction capabilities existing in the SDF utilizing flight software and functionally equivalent hardware simulating Spacelab Command and Data Management System (CDMS) failures.

Off-Nominal Training

Demonstration of off-nominal training in a flight software environment required inserting malfunction code into an existing experiment model which runs in the SDF host. The Wide Field Camera (WFC) model was selected for this demo because it has an Experiment Computer Applications Software (ECAS) display, an ECAS task, and the level of fidelity of the existing model was nearer that required for demonstration purposes than other SpaceLab models existing in the SDF.

No significant problems were encountered during the preparation for this demonstration. No special changes were made to the experiment model software other than malfunction insertion. Also, no changes were made to the flight software. Normal ECAS responses to trainee inputs via the display interface were utilized and demonstrated.

Conclusions reached from this demonstration are two-fold:

1. A high percentage of commonality and excellent potential for synergism exists between systems used for software development/verification and training.

2. The augmentation required for training on a system used for software development/verification was demonstrated to be feasible with a minimal amount of effort.

The following additional considerations/implications are evident from the demonstration:

1. Using flight software can facilitate the use of actual (e.g. prototype, engineering model) payload hardware for training.

2. Flight software usage eliminates redundant updates to simulators which emulate the flight software.

Validation Automation

The Validation Automation portion of the demonstration is relevant to the SCS requirements for automatic scripts, trainee absent mode, automated activation, and automated test support. Validation automation, or VALAUTO as it is called in the SDF, is a capability which allows hands-off generation and execution of test cases designed to test the different signal interfaces defined in the experiment computer database. This capability was incorporated into the SDF approximately two years ago, and has proven to be a worthwhile investment. VALAUTO consists of:

a.) a test case file which consists of operator commands and keystrokes,

b.) test controller software on the SDF host that executes the test command file, captures test results from the downlink, and performs test recording and initial analysis activities,

c.) an application task on the flight computer that emulates keyboard inputs.

Presently in the SDF, VALAUTO is only utilized for testing, however, the potential for the use of this concept is much greater. The benefits and other capabilities of VALAUTO are:

• Provides automated testing/operation without on-line operator/user

-- Represents trainee-absent mode

• Allows exact repeatability

-- Makes regression testing easier

-- Provides capability for operations evaluation functions

• Saves time and money

-- Frees personnel and computer resources to perform other tasks

-- Allows training with some trainers not staffed (e.g., stand-alone POIC training)

• Provides the capability for an entire increment to be tested in batch mode

-- Whole test case file may be run overnight

The functionality of validation automation would be very easy to implement if included in the early stages of SCS design. Validation automation is a complex task written in Perkin Elmer assembler and for development at the SDF required much time and investigation into the system and a great deal of effort, however, the effort would have been reduced considerably if the SDF had been designed to evolve this capability. The SDF validation automation software was not modified for the demo.

Generic Malfunctions

A demonstration of malfunction capabilities utilizing Spacelab flight software and the functionally equivalent flight hardware available in the SDF was also presented. This demonstration provides insight into functionality that may not be available in a simulated flight software and hardware environment. By simulating CDMS failures, the following three malfunction areas were demonstrated:

1.) Simulation of a Mass Memory Unit (MMU) failure -- Upon trying to load an ECAS task, a MMU error occurred because the MMU was simulated to be dead. The Spacelab MMU performs an equivalent function to the Space Station Mass Storage Unit (MSU).

2.) Simulation of a Remote Access Unit (RAU) failure -- Via simulation, RAU 5 stopped responding which caused non-operational replies to be received for the UIT and HUT experiments and also generated many error messages (e.g., RAU SKIPPED) on the fault summary page. The Spacelab RAU is equivalent to the Space Station Multiplexor/ Demultiplexor (MDM).

3.) Change of System Time -- The flight simulation time was changed from the simulation control station. The variance in the simulated

Master Time Unit (MTU) signal caused Experiment Computer Operating System (ECOS) to switch to the flight computer's Real Time Clock (RTC) and generated appropriate error messages on the fault summary page. A RTC to MTU switch was initiated from the ECOS DPM page to resynchronize the simulation time and flight computer time. The Spacelab/Orbiter MTU is equivalent to the Space Station time distribution system.

A key point from this demonstration was the automatic failure propagation that is inherent in flight software based systems. Simulated failures of various components of the Spacelab Computer Data Management System (CDMS) caused realistic crew responses without complicated integrated cause and effect models. When the RAU was caused to fail, not only were appropriate error messages displayed by ECOS, but ECOS also did not activate the appropriate experiments.

Significant Design-based Demonstration Conclusions

The demonstration showed the feasibility and positive impacts of using a flight software based trainer. For the development of a flight software based SCS some additional considerations must be addressed. Flight software may not be available or practical for all training applications so the integration of DMS and non-DMS trainers should be considered. Additionally, experiment Principal Investigators (PIs) must have access to development and testing capabilities for flight software interface simulations. To ensure successful reuse of software development simulations in the trainer, training requirements must be incorporated into the original simulation requirements.

The considerations noted above are chiefly programmatic issues. The demonstration raised no technical roadblocks to the design of this type of trainer. In fact, based on the relatively small modification of the WFC model for the demonstration, it is evident that training requirements could be included in the original model development with minimal impact. Therefore, the SCS design should address capabilities for use of flight software test and verification simulations. At a programmatic level, a mechanism for levying training requirements on the development of both system and payload simulations must be identified and pursued.

Generic//Programming Panels and Rapid Prototyping

This demonstration explored the potential use of graphics hardware, touch screens, and rapid prototyping software for the SCS training. Two main factors are of interest:

a) How much can a rapid prototyping product or virtual screen product improve the productivity of SCS simulation developers?

b) Can current technology produce realistic enough pictures and interaction with a prototyped experiment front panel to be usable in the SCS?

The demonstration was presented by the Virtual Prototypes Company of Dayton, Ohio. They have developed software and a small amount of hardware into a commercial prototyping product name VAPS. The VAPS software runs on Silicon Graphics and Megatek workstations currently. Before the end of the year, the VAPS software will also run on the HP 9000 and DEC workstations. In 2 years, it will also run on the high end Sun workstations..

The VAPS software does indeed permit quick prototyping of front panels. During the demonstration, a panel consisting of two CRTs, three buttons, one rotating knob, and two rotating compass roses was constructed from scratch in a few minutes. Additionally, VAPS will produce both an ASCII plain text description of the commands needed to produce the graphic picture, and also a C language version of the picture. However, these are in Silicon Graphics display primitives, and not in one of the graphic standard set of primitives. Nevertheless, it was clear that this tool would increase simulation development by some significant factor.

Several of the displays clearly demonstrated the realism of the graphics that could be produced. One, the F-18 avionics radio, looked very much like a picture of the unit instead of something computer generated. The high resolution touch screen used in the demonstration provided a very good interaction with the F-18 panel. Although no consensus was asked for or reached, it seemed that this product provided about as much realism as might be obtainable from a virtual panel.

CONCLUSIONS

The SCS study is based on the 1 August 1989 SSFP baseline, thus the results do not reflect the currently underway configuration/budget review (C/BR). As a result of the many design uncertainties in the DMS Kits, SIB, and SSE, the range of potential SCS designs to be considered remains quite high. At this time, none of the three SCS Task 5 refined designs can be either selected or discarded.

In spite of all the uncertainties, the SCS study effort has resulted in three viable refined design alternatives which can be used as a beginning baseline and for comparisons in any future SCS design work. Most importantly, the SCS Study has produced a thorough and well thought out set of SCS candidate requirements.

The uncertainty in SSFP elements also means that it may be possible for NASA MSFC PTC/SCS personnel to utilize the results of the SCS Study to influence the DMS Kit, SIB, and SSE requirements to support a better, flexible, and more efficient PTC/SCS design.

Toward this end, the following recommendations are made:

- The SCS design work indicates that if the SIB could connect to a network, rather than a mainframe host, far greater SCS design flexibility would result.

- The DMS Kits and SIB must support simulation control, including start, stop, reset, freeze, data store for later restart, restart from stored data, step ahead, and synchronization of processors with simulations.

- DMS Kit and SIB design must support real-time responses.

- DMS Kit and SIB design must support some level of malfunction capability.

- DMS Kit design must duplicate the on-board configuration, including simulation of both core and payload networks, and use of operational flight software.

- Maximum utilization of flight software test and verification simulations by SCS is recommended. This means that the training designers must have a voice in the flight system simulation requirements. It also implies that the architectures be similar enough to allow use of flight system simulations for training purposes.

- Some standard way of connecting C&D panels must be utilized to help ensure transportability of simulations between training facilities.

- Some standard simulation architecture and control method must be followed to minimize integration time and help ensure transportability of simulations between training facilities. This is especially important for simulations transported in both directions between the PTC/SCS and the SSTF.

- Utilization of the DMS Kits is recommended for the PTC/SCS. These will ease the computing burden considerably and help ensure flight equivalent payloads can be used with the SCS.

- The standardization on Ada and SSE is recommended. Though a front end burden, it will pay dividends over the lifetime of the SSFP.

- Utilization of the commonality and synergism between flight software verification capabilities and payload crew training capabilities (as demonstrated in the design-based technology demonstration) is recommended.

- Follow up investigation of the potential of virtual panels for use in the PTC/SCS is recommended. The virtual panel technology demonstration identified virtual panels as potential key technology.

In view of the current SSFP and SCS programmatic uncertainties and schedule slips, the following strategies are suggested for the future of SCS:

- Keep abreast of changes in the status of the DMS Kits, SIB, and SSE. These will have large effects on the SCS requirements and design.

- Track the SSTF design to ensure transportability of simulations in both directions.

- Continue to track the various technologies relevant to SCS during the projected slip in the SCS development schedule.   Some significant breakthrough during this period could affect the entire SCS approach.

- Most important of all, use the current projected slip in the SCS development schedule to continue to improve and update the candidate requirements. Stable requirements throughout the development phase are the best risk reduction tool for building a computerized system.

ASSUMPTIONS SUMMARY

The Task 1 Study Issues Assumptions were made from August 1988 to November of 1988 during Task 1. The assumption that the use of virtual instruments or panels would only be usable in part task trainers was changed by the technology demonstration on virtual panels (Study Issues Assumption #26). Virtual panels may have a much broader use, possibly even in the consolidated trainer.

The Task 2 Study Analysis Assumptions were made from October 1988 to March 1989. An original assumption that 20% of payloads would be changed out in every increment has since been changed to a worst case change out of 15% per increment (Study Analysis Assumption #11).

TASK 1 - Study Issues Assumptions

1.   Primary responsibility of the PTC is to provide payload operations training including both nominal and contingency operations for flight and ground personnel.

2.   Payload operations training will include experiment training, payload unique operations support systems training , and payload unique subsystems training.

3.   The PTC/SCS will support all manned payload training for all payloads, including US Lab, Attached Payloads, Columbus, and JEM.

4.   ESC training is assumed to not be a responsibility of the PTC. No unique interface between the ESC and the SCS is required, nor will the SCS simulators require any unique capabilities related to ESC training. The only support of the SCS to ESC training will be via the POIC during consolidated and/or integrated simulations.

5.   Any PTC interfaces to UOFs, DOCs, or ROCs will be through the POIC. There will not be any direct data interfaces from the PTC to the UOFs, DOCs, or ROCs.

6.   The POIC can support the processing of real time or simulated data streams simultaneously. This means the POIC can support training using simulated data from the PTC simultaneous with on going real time operations.

7.   The OMS software functions will be provided as part of the DMS Kits. Therefore no special simulator development will be required for OMS training.

8.   The PTC will not be responsible for any subsystems training. However, the PTC will utilize minimum subsystem interfaces as necessary to support payload training.

9.   All software subsystem simulators utilized in the PTC will be provided by work package contractors via the SSE. Modifications of these will be required.

10.   PTC experiment simulators will only provide high fidelity simulation of the housekeeping data. Experiment science data will not be dynamically simulated.

11.   For loading purposes, all simulations are assume to be done via software. However, the PTC/SCS is assumed to have the hooks and scars to support flight equivalent hardware and software when it is available.

12.   The PTC will utilize DMS Kits that are provided by JSC/WP02.

13.   The PTC will provide for the generation of all experiment data stream formats including dedicated experiment channel data streams. However, the data to fill dedicated experiment data streams will not be dynamic.

14.   All data on the payload bus will be simulated at the PTC. The payload bus includes two nodes with 10 megabits of data on each node. The PTC shall also output the data from the systems bus which also contains 10 megabits.

15.   Experiment prototype systems will be able to interface to the PTC data stream generator to provide dedicated experiment channel data.

16.   ECWS simulators will be provided by WP01 contractor.

17.   The PTC will be required to support full consolidated experiment operations training on 3 SS increment configurations simultaneously (2 U.S. Labs and 1 Columbus/JEM) with part task training on individual experiments from 3 other increments (each of the 3 roughly equal to 1/3 of the U.S. Lab in capability).

18.   Development and verification efforts must be able to proceed simultaneously with training.

19.   For purposes of this study, training and development are assumed to be accomplished on a 40 hours per week day shift basis, with other hours reserved for backup, PM, and overflow work..

20.   A backup interface capability will be required between the PTC and the SSTF in order to execute payload simulators in the PTC in support of integrated simulations at the SSTF in some cases where it is not feasible to transport the payload simulator to the SSTF.

21.   Training done via remote execution is done on the SCS, or the trainees come to MSFC and train here.   Thus, the computing load would be the same, and will be accounted for in the study.

22.   No SCS simulation of EVA or SCS production of other rendered outside attached payload pictures.

23.   The standard update rate (required to support realistic displays for the trainees in the PTC) for the SCS for dynamic data will be once per second.   A subset of the simulator tasks (required to support realistic input by the trainees) will be required to execute at up to 10 Hz rate (e.g. response to hand controller inputs).   A rate of 25 Hz may be required for pointing systems.

24.   To work with the core subsystem flight equivalent hardware and software, the SCS must work at rates that satisfy this flight equivalent hardware and software.

25.   Onboard data storage capability of the DMS will be part of the DMS simulation capability provided by JSC/WP02.

26.   Virtual instruments are acceptable in the part task experiment simulation workstations, but may not be good enough for use in the consolidated increment training environment.

27.   There are currently no requirements for onboard training levied on the SCS.

28.   SCS will use SSE capabilities for software development and maintenance.

29.   For purposes of loading for the SCS study, all simulations are assumed to be built, integrated, tested, and maintained on the PTC/SCS. Sizing must be done for worst case.

30.   Late changes to the simulators are a problem that the PTC/SCS people have to solve.

31.   Recent top level agreements that the crew will train together at JSC the final period before launch dictate that the SCS simulations will be transportable to the SSTF.

32.   The PTC/SCS people at MSFC will be responsible for maintenance of the payload simulators, including the period when the simulators are used at the SSTF.

33.   Integration of the payload simulators into the SSTF will be through a JSC/MSFC agreed upon method.

34.   There must be a capability to simulate payloads, if only for simulator maintenance, at the PTC/SCS for the duration of the payload's mission life. Thus, if a payload simulator is both hardware and software, the hardware may be duplicated, virtual panels may be used, or a parallel software simulator may be developed in order to retain the payload simulation capability at the PTC/SCS. The duplicate/substitute simulator may be employed at JSC in place of the original hardware/software simulator at the PTC.

35.   The SCS will support payload flight hardware, i.e. the SCS will have the hardware, software, hooks, and scars to support flight equivalent payload hardware.

36.   Assume the Space Station life cycle is 30 years, but that computers, displays, and other COTS electronic equipment will have to be replaced or upgraded at intervals ranging from 5 to 10 years.

37.   A host-based DMS functional simulation (FSIM) to be provided by SSE will NOT run in real time. Thus, FSIM is assumed to be of minimal utility in the SCS.

38.   The current assumption is that all interfaces to SSIS and SOAN will be through the POIC. Thus, the SCS need only worry about the proper interface to the POIC, and the POIC will solve further external interface problems.

TASK 2 - Study Analysis Assumptions

1.   A worst case load on the SCS is assumed for payload models which might be simulated by either physical hardware or a simulator executing on the PTC SCS.

2.   Code size estimates are based on Spacelab experience using structured analysis and FORTRAN coding.   These estimates may be reduced using new CASE tools and new programming methodologies.

3.   OMS functionality with respect to the onboard application (OMA) will be provided by the flight software resident on the DMS kits. OMS functionality with respect to the ground application (OMGA) will not be a part of the DMS and will have to be provided by other means.

4.   A worst case load on the SCS is assumed for subsystem models which might be required to support payload training.

5.   The SCS ECWS is not required to provide the nonpayload related functionality of the flight equivalent ECWS. However, the flight equivalent ECWS may be used for the SCS ECWS.

6.   The SCS video system will be compatible with the video subsystem of the SS Communications and Tracking system.

7.   The SCS video system will be capable of interfacing to the SSIS network.

8.   The SCS video system, while compatible with the SS video subsystem, is not required to provide 100% of the functionality of the SS video subsystem.

9.   SCS video data is not required to be digitized and sent through the C&T system to the POIC.

10.   Late changes to the simulators are a problem that the PTC/SCS people have to solve.

11.   Payload change out per increment will be a worst case of 15% per increment.

12.   Payload simulations are software simulations 90% of the time.

13.   Development will be done concurrently with training on the day shift. Swing and midnight shifts will be reserved for PM, CM, and backups.

14.   The PTC will not be responsible for any subsystems training. The PTC will utilize the minimum subsystem interfaces required to support payload training.

15.   The PTC will use NASA-provided DMS kits.

16.   A host-based DMS functional simulation (FSIM) to be provided by SSE will not run in real time. Thus, FSIM is assumed to be of minimal utility in the SCS.

GLOSSARY

BNIU - Bus Network Interface Unit
CBT - Computer Based Training
CDMS - Computer Data Management System
CM - Configuration Management
CMDM - Control Multiplexer/Demultiplexer
COTS - Commercial Off The Shelf
CPU - Central Processing Unit
CSC - Computer Software Configuration
CSCI - Computer Software Configuration Item
CUI - Common User Interface
DFD - Data Flow Diagram
DID - Data Item Definition
DMS - Data Management System
DOC - Discipline Operation Center
DOD - Department of Defense
DPM - Display Memory
EBCDIC - Extended Binary-Coded Decimal Interchange Code
ECAS - Experiment Computer Application Software
ECOS - Experiment Computer Operating System
ECWS - Element Control Workstation
EDP - Electronic Data Processor
ESA - European Space Agency
ESC - Engineering Support Center
EVA - Extravehicular Activity
FCD - Functional Control Document
FEL - First Element Launch
FMPAC - Fixed Multi-Purpose Applications Console
FSIM - Functional Simulation
GFE - Government Furnished Equipment
GSE - Ground Support Equipment
GSFC - Goddard Space Flight Center
HWCI - Hardware Configuration Item
Hz - Hertz
IT&V - Integration, Test, and Verification
JEM - Japanese Experiment Module
JSC - Johnson Space Center
LAN - Local Area Network
MDM - Multiplexor/Demultiplexor
MEA - Mass Energy Analysis
MIPS - Millions of Instructions per second
MMU - Mass Memory Unit
MPAC - Multi-Purpose Application Console
MPS - Mission Planning System
MSFC - Marshall Space Flight Center
MSU - Mass Storage Unit

MTBF - Mean Time Between Failure
MTTR - Mean Time To Repair
MTU - Mass Timing Unit
NASA - National Aeronautics and Space Administration
NDT - Non-scheduled Downtime
OMA - Operation Management Application
OMGA - Operation Management Ground Application
OMS - Operation Management System
OOD - Object Oriented Design
PC - Personal Computer
PCTC - Payload Crew Training Complex
PD - Payload Developers
PFE - PTC Facility Equipment
PI - Principal Investigator
PM Preventive Maintenance
PMMS - Process Materials Management Subsystem
PMPAC - Portable Multi-Purpose Applications Console
POGA - Payload Operations Ground Application
POIC - Payload Operations Integration Center
PTC - Payload Training Center
PTD - PTC Training Devices
PTT - Part Task Trainer
RAU - Remote Acquisition Unit
ROC - Regional Operation Center
RTC - Real Time Clock
SA - Structured Analysis
SCS - Simulation Computer System
SDF - Software Development Facility
SDP - Standard Data Processor
SDT - Scheduled Downtime
SIB - Simulation Interface Buffer
SOAN - Science Operations Analysis Network
SPA - System Product Assurance
SRT - Scheduled Application Run Time
SS - Space Station
SSCC - Space Station Control Center
SSE - Software Support Environment
SSF - Space Station Freedom
SSFP - Space Station Freedom Program
SSIS - Space Station Information System
SSP - Space Station Program
SSTF - Space Station Training Facility
TBD - To Be Determined
TGU - Time Generation Unit
TMIS - Technical Management Information System
UIMS - User Interface Management System
UOF - User Operation Facility
USE - User Support Environment
VALAUTO - Validation Automation

VAPS - Virtual Prototyping
WFC - Wide Field Camera

Report Documentation Page

NASA

1. Report No	2. Government Accession No.	3. Recipient's Catalog No.
N/A		

4. Title and Subtitle	5. Report Date
Simulation Computer System (SCS) Study Report	October 1989
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
	N/A
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No.
TRW, Grumman, Essex	NAS8-37745
	13. Type of Report and Period Covered

12. Sponsoring Agency Name and Address	Study Report Aug 88 - Oct 89
NASA MSFC EL12 Huntsville, AL 35812	14. Sponsoring Agency Code

15. Supplementary Notes

NASA Contact	-	Michael Watson, MSFC, EL12
TRW	-	Jeff Jensen, 213 Wynn Drive, Huntsville, AL 35805
Grumman	-	Michael Denny, 1000 Woodbury Road, Woodbury, NY 11797
Essex	-	Dan Dodson, 690 Discovery Drive, Huntsville, AL 35806

16. Abstract

NASA's Space Station Freedom program (SSFP) planning efforts have identified a need for a payload training simulator system to serve as both a training facility and as a demonstrator to validate operational concepts. The envisioned MSFC Payload Training Complex (PTC) required to meet this need will train the Space Station payload scientists, station scientists, and ground controllers to operate the wide variety of experiments that will be onboard the Space Station Freedom. The Simulation Computer System (SCS) is the computer hardware, software, and workstations that will support the Payload Training Complex at MSFC.

The purpose of this SCS Study is to investigate issues related to the SCS, alternative requirements, simulator approaches, and state-of-the-art technologies to develop candidate concepts and designs.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Space Station Payload Training Simulator Computer	

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
UNCLASSIFIED	UNCLASSIFIED	6 vols, 40-100 ea.	

NASA FORM 1626 OCT 86